

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Факультет інформатики та обчислювальної техніки
Кафедра автоматики та управління в технічних системах**

До захисту допущено:

Завідувач кафедри

_____ Олександр РОЛІК

«__» _____ 20__ р.

Дипломний проєкт
на здобуття ступеня бакалавра
за освітньо-професійною програмою «Комп'ютеризовані системи управління»
спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології»
на тему: «Автоматична система генерації запитів до бази даних на основі
асинхронних операцій»

Виконав (-ла):

студент (-ка) IV курсу, групи ІА-61

Тюрін Валерій Олександрович _____

Керівник:

к.т.н., доц. Савчук О.В. _____

Консультант з: _____

Рецензент:

Проф. каф. ОТ НТУУ

«КПІ ім. Ігоря Сікорського», д.т.н.

Сімоненко Валерій Павлович _____

Засвідчую, що у цьому дипломному проєкті
немає запозичень з праць інших авторів без
відповідних посилань.

Студент (-ка) _____

Київ – 2020 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки

Кафедра автоматики та управління в технічних системах

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 151 «Автоматизація та комп'ютерно-інтегровані технології»

Освітньо-професійна програма «Комп'ютеризовані системи управління»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Олександр РОЛІК

« ____ » _____ 20__ р.

ЗАВДАННЯ

на дипломний проєкт студенту
Тюріну Валерію Олександровичу

1. Тема проєкту «Автоматична система генерації запитів до бази даних на основі асинхронних операцій», керівник проєкту к.т.н., доц. Савчук О.В, затверджені наказом по університету від «07» травня ____ 2020р. №1081-с

2. Термін подання студентом проєкту _____

3. Вихідні дані до проєкту

4. Зміст пояснювальної записки

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо)

6. Консультанти розділів проєкту*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

* Якщо визначені консультанти. Консультантом не може бути зазначено керівника дипломного проєкту.

7. Дата видачі завдання 12.03.2020

Календарний план

№ з/п	Назва етапів виконання дипломного проєкту	Термін виконання етапів проєкту	Примітка
1	Затвердження теми роботи	12.03.2020	
2	Аналіз існуючих рішень	26.03.2020	
3	Опрацювання літературних джерел пов'язаних з асинхронними операціями	10.04.2020	
4	Розробка функціональної блок схеми	17.04.2020	
5	Розробка архітектури проєкту	24.04.2020	
6	Розробка бібліотеки	08.05.2020	
7	Розробка тестового застосунку	13.05.2020	
8	Оформлення матеріалів проєкту	22.05.2020	
9	Передзахист	25.05.2020	
10	Захист	15.06.2020	

Студент

Валерій Тюрін

Керівник

Олена Савчук

АНОТАЦІЯ

Тюрін В.О. Автоматична система генерації запитів до бази даних на основі асинхронних операцій. КПІ ім. Ігоря Сікорського, Київ, 2020.

Проект містить 79 с. тексту, 56 рисунків, 1 таблицю, посилання на 23 літературне джерело, 4 додатків.

Ключові слова: асинхронний запит, база даних, AWS, Spring, ClassLoader, DynamoDB, хмара, неблокуюча операція, I/O, навантаження.

Об'єктом розробки є автоматизована система генерації запитів до бази даних на основі асинхронних операцій.

Мета розробки – підвищення швидкості опрацювання конкуруючих запитів та скорочення ресурсів на підтримку застосунків.

Методи дослідження: абстрагування, аналіз та синтез, індукція та дедукція, моделювання, класифікація, конкретизація, аналіз та вибір технологій та інструментів, порівняльний аналіз, аналіз результатів, обробка літературних джерел, аналіз сценаріїв, тестування, маркетинговий та економічний аналіз, економіко-математичне моделювання, вимірювання основних показників, експеримент.

Отримані результати будуть корисними для практичного застосування в високонавантажених системах орієнтованих на хмарні рішення, в наукових цілях для вивчення характеристик систем, що побудовані на неблокуючих операціях та в навчальних цілях для вивчення можливостей асинхронного програмування на базі Java та Amazon Web Services.

Пояснювальна записка
до дипломного проєкту
на тему: «Автоматична система генерації запитів до бази
даних на основі асинхронних операцій»

Київ – 2020 року

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	3
ВСТУП.....	4
1 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ В ГАЛУЗІ ГЕНЕРАЦІЇ ЗАПИТІВ ДО БАЗ ДАНИХ.....	7
1.1 Призначення і галузь застосування	7
1.2 Аналіз існуючих рішень	8
Висновок до розділу 1	10
2 ОГЛЯД ЗАПРОПОНОВАНОГО РІШЕННЯ ТА ТЕХНОЛОГІЙ.....	12
2.1 Огляд рішення	12
2.2 Вибір технологій для реалізації проєкту	14
Висновок до розділу 2	21
3 РЕАЛІЗАЦІЯ БІБЛІОТЕКИ	23
3.1 Цикл ініціалізації бібліотеки.....	23
3.2 Опис процесу генерації запитів в запущеному застосунку	28
3.3 Опис використаних шаблонів проєктування в архітектурі	30
Висновок до розділу 3	32
4 СТВОРЕННЯ ТЕСТОВОГО ДОДАТКУ З МЕТОЮ ТЕСТУВАННЯ БІБЛІОТЕКИ.....	33
4.1 Опис принципів побудови схем нереляційних баз даних.....	33
4.2 Огляд можливих середовищ розгортання DynamoDB	35
4.3 Створення тестової бази даних.....	40
4.4 Створення тестового додатку	45
4.5 Опис технічної реалізації тестового додатку	48
4.6 Реалізація навантажувального тестування	54
4.7 Види блокування	73
Висновок до розділу 4	74
ВИСНОВКИ.....	76
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	78

					ІА61.260БАК.003 ПЗ			
			підпис	ДАТА				
СТУДЕНТ	ТЮРІН В.О.				Автоматична система генерації запитів до бази даних на основі асинхронних операцій			Аркушів
КЕРІВНИК	САВЧУК О.В.							79
							НТУУ «КПІ ім. Ігоря Сікорського», ФІОТ, ІА-61	

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

API	(Application Programming Interface) Інтерфейс прикладного програмування
HTTP	(Hyper Text Transfer Protocol) Протокол передачі гіпер-текстових документів
HTTPS	(Hyper Text Transfer Protocol Secured) Безпечний протокол передачі гіпер-текстових документів
RAM	(Random access memory) пам'ять із довільним доступом; оперативна пам'ять
REST API	(Representational State Transfer) підхід до архітектури мережеских протоколів, які забезпечують доступ до інформаційних ресурсів.
JRE	(Java Runtime Environment) середовище виконання Java
JDK	(Java Developer Kit) середовище розробки для Java
JVM	(Java Virtual Machine) віртуальна машина, що перетворює команди в машинний код та забезпечує платформи-незалежність
IAM	(Identity and Access Management) система розподілу прав доступу
GC	(Garbage Collector) механізм автоматичної чистки невикористовуваних об'єктів в високорівневих мовах програмування
БД	база даних
SDK	(Software Development Kit) набір певних інструментів, що допомагають стороннім розробникам створювати ПЗ інтегроване з певним пакетом програм
AWS	(Amazon Web Services) набір сервісів, що компанія Amazon надає як послугу
IoT	(Internet of Things) мережа, яка складається з пристроїв
I/O	(Input/Output) передача даних на вхід або вихід
VPC	(Virtual Private Cloud) віртуальна приватна хмара
Bean	(Бін) об'єкт класу, яким керує Spring IoC контейнер

					ІА61.260БАК.003 ПЗ	Арк.
						3
Змн.	Арк.	№ докум.	Підпис	Дата		

ВСТУП

Проект є Open Source бібліотекою для використання в проєктах, орієнтованих на хмарні сервіси та нереляційні бази даних. Використання клієнтом даної бібліотеки звільнить його від необхідності писати запити до бази даних – це стане роботою бібліотеки.

Крім того, застосування логіки, яка базується на відповіді з БД, буде виконуватись асинхронно, що дає конкурентні переваги над аналогами при багато поточності.

Наразі подібні системи обмежені у попиті, тому що, далеко не всі компанії використовують хмарні нереляційні сховища, через їх новизну та високу ціну, й через те, що асинхронні програми мають спеціалізоване призначення й підходять для дуже навантажених систем.

Системи, які базуються на асинхронних операціях будуть невпинно набирати популярність зі збільшенням загальної кількості інформації, й відповідно необхідності у системах, які будуть цю інформацію обробляти. В дипломному проєкті будуть детально описані та протестовані на реальній системі приклади, що продемонструють сильні та слабкі сторони цього підходу.

Розмір фінансових інвестицій та кількість наукових робіт в галузі систем, що базуються на неблокуючих операціях та використанні хмарної інфраструктури, невпинно зростають. В даній галузі є вже достатньо реалізованих систем та популярних наукових робіт в досить вузькій сфері систем, що засновані на повідомленнях й чергах, таких як Kafka. Даний дипломний проєкт освітлює тематику використання неблокуючих операцій у поєднанні з нереляційними БД. Слід зазначити, що розповсюдженість баз даних набагато вища, а практичні дослідження їх використання майже відсутні. Дипломний проєкт є інноваційним й освітлює на прикладі, ще неописані можливості неблокуючих систем заснованих на хмарних рішеннях.

					ІА61.260БАК.003 ПЗ	Арк.
						4
Змн.	Арк.	№ докум.	Підпис	Дата		

Для тестування і оцінки якості системи створений тестовий проєкт, в який інтегрована Open Source бібліотека. Даний проєкт є функціонально готовим до експлуатації.

Реалізація деяких рідковживаних сценаріїв заблокована насамперед через незавершений асинхронний клієнт Amazon SDK.

В дипломному проєкті будуть оглянуті існуючі рішення в галузі генерації запитів до баз даних й описані їх недоліки в порівнянні з розробленою системою.

Буде проведений детальний огляд неблокуючих операцій та сам процес блокування в ПЗ, що допоможе чітко визначити проблематику.

За допомогою механізмів, які надає мова Java, розглянуті неблокуючі операції будуть реалізовані.

Перед проєктом стоїть мета - це підвищення швидкості опрацювання конкуруючих запитів та скорочення ресурсів на підтримку застосунків.

Для цього потрібно розв'язати наступні задачі:

- безпомилково генерувати найчастіше вживані запити, такі як збереження, оновлення, отримання, видалення;
- надавати можливість асинхронно обробляти результати;
- мати можливість обробляти помилки;
- мати значну швидкодію та високу надійність при високих показниках конкуруючого доступу до ресурсів;
- легко масштабуватися.

Задля того, щоб реалізувати систему, яка відповідає заданій меті необхідно провести теоретичні та практичні дослідження.

Дослідження фактично були поділені на чотири етапи:

- базової теоретичної підготовки, що включає в себе теоретичну підготовку по всім архітектурним та технологічним напрямкам, таким як вивчення внутрішньої будови DynamoDB, інфраструктури AWS, Spring Framework та інших використаних технологій;

					ІА61.260БАК.003 ПЗ	Арк.
						5
Змн.	Арк.	№ докум.	Підпис	Дата		

- теоретичної підготовки в розділах асинхронних систем, що включає в себе покращення знань та навичок в реалізації неблокуючих операцій механізмами Java, вивчення та впровадження шаблонів проєктування рекомендованих для асинхронних систем;
- технічної реалізації проєкту;
- оцінки якості, що включає в себе автоматизоване та мануальне тестування всіх рівнів розробленого застосунку, що було реалізоване за допомогою написання додаткового ПЗ.

Дослідження, що були проведені при виконанні проєкту мають науковий потенціал:

- як українізований посібник для починаючих розробників в асинхронному програмуванні;
- як тестовий застосунок, що на реальних прикладах демонструє доцільне та недоцільне використання асинхронності.

Проєкт є безкоштовним для користувачів. Цільовим сегментом є проєкти, що мають велике навантаження на свої застосунки та використовують найпопулярнішу нереляційну хмарну БД – DynamoDB. Розроблений проєкт може запропонувати для них інтеграцію з новим асинхронним SDK від AWS, який підвищить швидкодію систем, а проста, проте виконана за шаблонами Java та Spring, архітектура дозволить легко інтегруватися та не витратити час на вивчення абсолютно нових технологій.

Для технічної реалізації ідей були обрані найновіші версії стандартизованих технологій. Новизна дає змогу використовувати найоптимальніші підходи для вирішення проблем, а використання стандартів перевірених часом робить систему надійною та легкою для розуміння для розробників, що працюють в цій сфері.

					ІА61.260БАК.003 ПЗ	Арк.
						6
Змн.	Арк.	№ докум.	Підпис	Дата		

1 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ В ГАЛУЗІ ГЕНЕРАЦІЇ ЗАПИТІВ ДО БАЗ ДАНИХ

1.1 Призначення і галузь застосування

Дипломний проєкт призначений для делегування роботи по побудові запитів до бази даних від застосунку до бібліотеки, підвищення якості та зменшення витрат на підтримку клієнтських застосунків. Бібліотека призначена для роботи під великим навантаженням та конкуренцією між потоками за спільні ресурси. Бібліотека здатна генерувати найчастіше вживані операції при роботі з базою даних, та надавати можливість асинхронної обробки результатів та помилок виконання. Бібліотека призначена виключно для хмарних рішень, що базуються на AWS. Застосунок може працювати в локальному режимі, тобто без хмари, однак, цей режим виключно для розробників та тестувальників, а не користувачів.

Галуззю застосування є хмарні рішення, що мають event-driven архітектуру, та опрацьовують результати базуючись на методах Publisher-Subscriber. Бібліотека може бути використана і при інших архітектурних підходах, однак, вона не зможе в інших умовах показати кращі показники за часом ніж синхронні аналоги.

Бібліотека ідеально вписується у сервіси, які будуть мати велику кількість функціонально не блокуючих малих операцій, прикладами можуть бути IoT застосунки, Service-to-Service застосунки. В окремих випадках бібліотеку рекомендовано використовувати й з такими програмами, що взаємодіють з людьми, якщо логіку взаємодії можна буде реалізувати, так, що користувач не буде повинен обов'язково чекати відповіді, та кількість таких користувачів буде надзвичайно великою(сотні тисяч за хвилину).

Дана бібліотека показує свої найсильніші сторони в event-driven застосунках.

					ІА61.260БАК.003 ПЗ	Арк.
						7
Змн.	Арк.	№ докум.	Підпис	Дата		

1.2 Аналіз існуючих рішень

AWS SDK – це засіб для усіх стадій розробки, тестування та управління програмним забезпеченням, орієнтованим на виконання в хмарі AWS. Найпростішим варіантом роботи з AWS DynamoDB – це використання клієнту, який надається AWS SDK для цієї БД, як на рисунку 1.1.

```
@PostMapping("/user/sync/create")
public ResponseEntity<Boolean> createUserSync(@RequestBody User user) {
    Map<String, AttributeValue> userAttributes = Map.of( k1: "UUID", AttributeValue.builder().s(user.getUuid()).build(),
                                                         k2: "USERNAME", AttributeValue.builder().s(user.getUsername()).build());

    return ResponseEntity.ok(userServiceSync.saveUserSyncClient(
        tableName: "PROD_USER", userAttributes).sdkHttpResponse().isSuccessful());
}
```

Рисунок 1.1 – Приклад формування об'єкту запиту

Головна перевага цього методу – це простота первинної реалізації. При використанні клієнту, вам не потрібні додаткові інтеграції, окрім тих, що ви і так маєте.

Очевидним недоліком цього підходу є необхідність конвертації об'єкту мови програмування, в нашому прикладі – об'єкт User, в запис в базі даних. Для кожного поля необхідно призначити назву, вказувати ім'я таблиці, тощо.

Така система генерації приводить до необхідності змішування утиліти конвертації з існуючим бізнес-кодом або об'єктами, що погіршує якість коду та робить тяжчим внесення змін. Змушує вводити додатковий DAO рівень.

Змушує підтримувати кожен створений унікальний варіант запиту. У складних системах може бути сотні або тисячі об'єктів і таблиць с десятками можливих операцій по кожній, таким чином необхідно підтримувати десятки тисяч запитів власноруч.

Spring Data – це частина Spring Framework, що робить легшим доступ та маніпуляції над даними.

Spring Data є одним з найширше використовуваних модулів Spring, що дає:

					ІА61.260БАК.003 ПЗ	Арк.
						8
Змн.	Арк.	№ докум.	Підпис	Дата		

- рівень абстракції над рівнем репозиторія та об'єктів;
- динамічне створення запитів;
- можливості аудиту;
- можливість додавання нових типів репозиторіїв;
- легку конфігурацію анотаціями або XML;
- потужну інтеграцію з модулем Spring MVC;
- підтримку одразу декількох сховищ [1].

Spring Data є раціональним вибором для більшості сервісів, в неї є лише 3 серйозні недоліки це:

- відсутність підтримки хмарних баз даних;
- оптимізація саме під синхронну модель роботи;
- велику кількість операцій, які використовуються в край обмеженому кількості випадків, але ускладнюють вивчення системи.

Boostchicken/spring-data-dynamodb - це Open-Source проєкт одного розробника з Каліфорнії [2], який адаптував Spring Data під блокуючий AWS DynamoDB client. Проєкт містить в собі переваги Spring Data та можливість роботи с AWS DynamoDB, однак близько двох років тому він перестав підтримувати проєкт.

Недоліки проєкту:

- наявність багів;
- неможливість робити деякі зміни та отримувати винагороду без згоди автора, який перестав цим займатись;
- надлишковість архітектури;
- оптимізацію під застарілі версії Java;
- використання старого AWS SDK.

На просторах інтернету є значна кількість невдалих спроб створити фреймворк для динамічної генерації асинхронних запитів. Усі вони є технічно недосконалими, тому що, автори намагалися отримати велику вигоду наслідуючи Spring Data, для зменшення кількості власної роботи,

					IA61.260БАК.003 ПЗ	Арк.
						9
Змн.	Арк.	№ докум.	Підпис	Дата		

однак, вона не пристосована та потребує дуже багато змін, щоб ефективно працювати асинхронно, тому, буде простіше відмовитися від архітектури Spring Data, і створити власну змінену, одразу орієнтовану під асинхронні операції.

Основні недоліки існуючих рішень:

- намагання пристосувати архітектуру Spring Data під те, для чого вона не створена;
- помилки в роботі з асинхронними запитами на рівні Java;
- відсутність теоретичної та практичної підготовки, що призвело до неправильного системного дизайну;
- невиправдана складність архітектури;
- нерозуміння області застосування асинхронних операцій і як наслідок неправильні ідеї та тестування.

Висновки до розділу 1

Проведений аналіз існуючих рішень показав, що:

- на ринку починає виникати попит на асинхронні системи, через загальне збільшення обсягу даних;
- пропозиція з боку ПЗ в цьому сегменті є обмеженою та слабо-розвиненою.
- для нових систем, що проєктуються під цю галузь - простота, швидкодія, надійність, масштабованість є ключовими чинниками;
- нові системи мають орієнтуватися на підтримку хмарних рішень через їх невпинне поширення;
- система має бути Open Source для отримання підтримки розвитку від широкої аудиторії.

					ІА61.260БАК.003 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

В проаналізованих рішеннях були знайдені такі недоліки:

- занедбаність проєкту через рекомендацію AWS не використовувати асинхронний клієнт в системах для кінцевих користувачів, так як над ним ще продовжують працювати;
- неправильний системний дизайн бібліотек;
- відсутність підтримки створених рішень;
- відсутність можливостей інтеграцій з розробленою системою;
- невиправдана складність архітектури.

					ІА61.260БАК.003 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

2 ОГЛЯД ЗАПРОПОНОВАНОГО РІШЕННЯ ТА ТЕХНОЛОГІЙ

2.1 Огляд рішення

Дипломна робота є Open Source проектом, був розроблений для заміщення системи описаної в огляді рішень підтримкою якої доводилося займатися.

Перше, що необхідно розуміти розглядаючи рішення – це те, що такі застосунки є ефективними виключно при застосуванні в областях описаних в п.1, та при правильних інтеграціях з іншими компонентами програм орієнтованих на хмару. Неблокуючі операції на рівні взаємодії з базою даних не дадуть жодного реального виграшу, якщо на рівні взаємодії з інтерфейсом користувача все одно потрібно буде дочекатися всього результату.

На блок-схемі зображено життєвий цикл проекту [IA61.260БАК.006 Д1].

Процес №1 показує спосіб інтеграції клієнтського застосунку і бібліотеки. Код бібліотеки збирається у .jar файл з вказуванням організації виробника – com.tiurINVALery, назви модуля та версії. Після цього будь-який застосунок, може зручно імпортувати бібліотеку як залежність.

Процес №2 відповідальний за підготовку метаданих класів на етапі запуску клієнтського застосунку. Перш за все, з ресурсу налаштувань проекту(це може бути файл, URL адреса файлу або програма, така як consul) бібліотека достає налаштування, в якому розробником клієнтського застосунку вказано назву головного пакета класів, для яких мають бути створені об'єкти мета-дати.

Після цього, для класів що знаходяться в цьому пакеті або в пакетах нижче за ієрархією, на етапі завантаження класів застосовується перевірка наявності анотації @Table – це означає, що треба створити для цього класу мета-дату для побудови запиті.

Процес №3 фактично є частиною логіки клієнтського застосунку, однак стосується бібліотеки. Звісно, в реальному середовищі бібліотека буде

					IA61.260БАК.003 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

працювати як складова частина більшої системи, що включає в себе інші модулі, ресурси та інфраструктуру, важливо мати на увазі те що, некоректна робота інших компонентів може впливати на якість та швидкодію роботи бібліотеки. Блок №3 - це абстракція логіки більшої системи.

Процес №4 є інтеграцією клієнтського застосунку з базою даних DynamoDB. На цьому кроці проходить підключення до БД, застосування IAM ролі (введення секретного accessKey, secretKey), перевірка наявності або створення схеми таблиць.

Процес №5 є процесом зміни стану проєкту з STARTING на RUNNING – перехід програми в робочий, готовий до запитів, стан. Перед переходом у цей стан, виконується перевірка на те, що сам застосунок з усіма обов’язковими компонентами готовий до роботи, а також усі інші ресурси, такі як БД та MessageQueue готові до роботи.

Якщо ця умова не виконується – застосується Fail-Fast підхід, та застосунок не перейде у стан RUNNING, замість цього він виведе детальний опис причин поломки.

Процес №6 є абстракцією алгоритму роботи клієнтського застосунку до запиту в БД.

Процес №7 є місцем побудови запиту. Першим кроком алгоритм визначається клас об’єкту, що має бути збереженим або отриманим з БД. Другим кроком дістається метадата цього класу, на її основі будується структура запиту та наповнюється даними з об’єкта. Після цього запит відправляється до БД. Якщо БД повідомляє, що прийняла запит, керування потоком передається клієнтському застосунку, в іншому разі – іде обробка помилки.

Процес №8 є абстракцією логіки обробки результатів БД запиту, та відправки нотіфікації до системи повідомлень.

Процес №9 є логікою роботи іншого ресурсу – MessageQueue, її суть в тому, що створюються папки для кожної окремої події, куди при виконанні 4.8 надсилаються повідомлення з первинно обробленим результатом з БД. На

					ІА61.260БАК.003 ПЗ	Арк.
						13
Змн.	Арк.	№ докум.	Підпис	Дата		

папку може бути підписаним 1 або більше Listener, який забирає потрібні йому повідомлення, та продовжує логіку роботи клієнтського застосунку.

Процес №10 є відповідальним за перехват та рішення виняткових ситуацій при роботі системи.

2.2 Вибір технологій для реалізації проєкту

Мова програмування Java була обрана для написання основної частини проєкту через те, що вона має найвище розповсюдження серед Back-End ПЗ та займає більшу частину Big Data застосунків [3]. Розподіл зображено на рисунку 2.1.

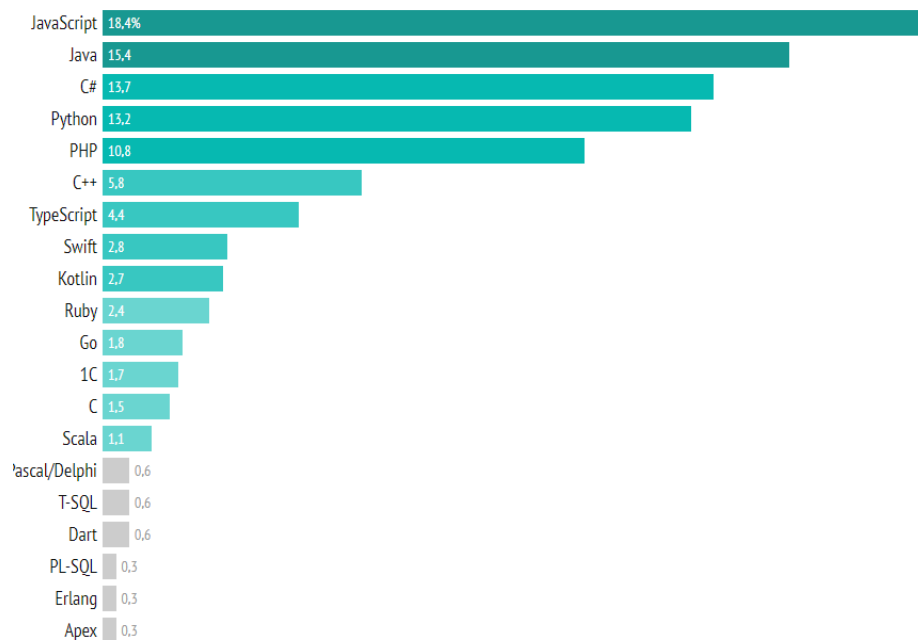


Рисунок 2.1 – Рейтинг популярності мов програмування

Java підтримується компанією AWS, яка навіть випускає свій JDK, що звісно сильно вплинуло на вибір.

Java надає сильні можливості розробнику для роботи з асинхронними операціями та з багато поточністю в цілому, що дуже важливо для бібліотеки, що працює на асинхронних операціях [4].

Java є мовою з найкращим рівнем можливості підтримки – це є особливо важливо в Open Source проєктах, тому що підтримувати бібліотеку доводиться людям з абсолютно різних команд, з різним рівнем знань технологій та дистанційно один від одного.

Java має найбільшу кількість якісних стандартів, завдяки яким архітектура та технічна реалізація є зрозумілою для нового розробника.

Java екосистема орієнтована на Open Source та розробку продуктів Community, на відміну від .NET середовища, що дозволяє легально використовувати безкоштовно найкращі застосунки для своєї програми.

Для проєкту була обрана Java 11, як найновіша версія з довготривалою підтримкою, на відміну від попередньої Java 8, вона надає можливість легко переходити на новіші версії у майбутньому та використовувати новий GC.

Spring – це основний сучасний Framework в Java екосистемі. Основною технологією, яку надав Spring з перших версій це Dependency Injection (DI далі). DI – вставка залежностей, це стиль налаштування об'єкта, коли поля об'єкту налаштовуються зовнішніми об'єктами. Окрім вставки залежностей, в проєкті також використані Spring Boot, Spring MVC.

Spring Boot спрощує написання мікросервісних Spring орієнтованих застосунків, які можна просто одразу запустити у будь-якому середовищі, де є JRE, завдяки вбудованому контейнеру. Spring Boot надає так звані стартери – класи, що виконують базові налаштування сервісу, що зменшує кількість конфігурацій зі сторони розробника, який мусить тільки змінити ті базові налаштування, які йому не підходять. Spring Boot надає додаткову функціональність у вигляді метрик, за допомогою яких дуже зручно можна одразу забезпечити моніторинг параметрів контейнера та додатку, перевірки стану сервіса, та можливість зовнішньої конфігурації.

Spring MVC шаблон проєктування, що розділює архітектуру застосунку, М – Model, V – View, С – Controller. Керуючі архітектурою слоїв застосунку, Spring робить застосунки стандартизованими, що добре впливає на розуміння системи, легкість підтримки та змін.

					ІА61.260БАК.003 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

Model рівень класичного Spring застосунку інкапсулює об'єкти додатку, це POJO та Bean's.

View відповідає за відображення даних моделі, як правило, генеруючі HTML.

Controller обробляє запити користувача і передає відповідну модель у візуалізацію, як видно на рисунку 2.2 [5].

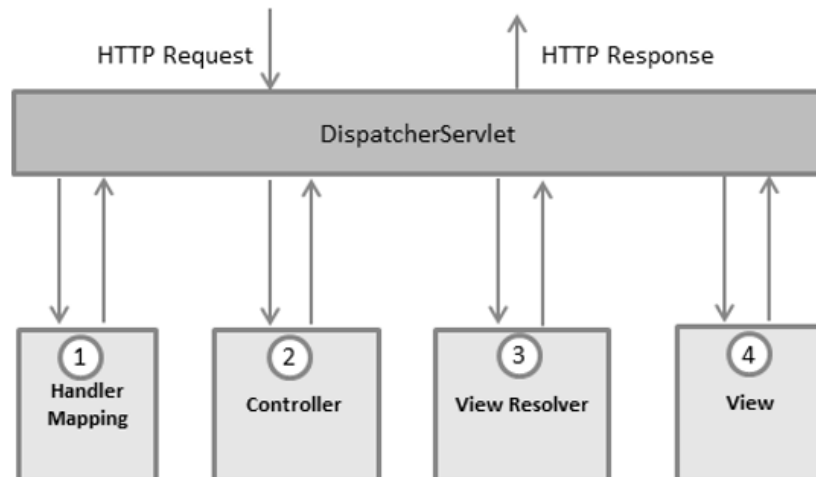


Рисунок 2.2 – Модель MVC

DynamoDB - це нереляційна ключ-значення та документна база даних з часом відповіді близькою до 1-10 мс [6,7].

DynamoDB забезпечує вбудовану підтримку транзакцій на стороні сервера, спрощуючи дії розробника по внесенню узгоджених змін в БД.

Поточне відновлення (PITR) допомагає захистити таблиці DynamoDB від випадкових операцій запису чи видалення. PITR забезпечує постійне резервне копіювання даних таблиці DynamoDB, і ви можете відновити цю таблицю в будь-який момент часу протягом попередніх 35 днів[6,7,8].

Переваги DynamoDB:

- вона автоматично та майже безмежно масштабована;
- підтримується AWS;
- захищена, з можливістю налаштування автоматичного backup;
- з рольовою системою IAM;
- з можливістю підтримки до 10 трильйонів запитів за добу.

DynamoDB використовується провідними компаніями світу, такими як Nike, Samsung, Netflix, Snapchat та багатьма іншими, саме тому бібліотека написана до цієї БД буде мати попит [7].

Docker - це програмна розробка для швидкої розробки, тестування і розгортання застосунків [9]. Завдяки Docker можна швидко розгорнути застосунок у будь-якому середовищі [9]. Використання Docker на AWS дає розробникам і системним адміністраторам надійний і економічний спосіб збирання, доставки і запуску розподілених додатків будь-якого масштабу. В основі роботи Docker лежить стандартизований спосіб виконання коду. Docker - це операційна система для контейнерів. Подібно до того як віртуальна машина створює віртуальне уявлення апаратного забезпечення сервера (тобто усуває необхідність безпосередньо управляти таким), контейнери створюють віртуальне уявлення серверної операційної системи [9].

В дипломному проєкті Docker використовувався для розгортання в ньому DynamoDB.

Це дозволило:

- зробити розробку та функціональне тестування безкоштовним;
- зробити розробку застосунку платформо-незалежною;
- розподіляти розробку та підтримку системи між багатьма розробниками, що можуть взяти код застосунку, запустити еквівалентну БД та проводити тестування або розробку власного додаткового функціоналу.

Apache Maven - це інструмент управління програмного забезпечення. Спираючись на концепцію моделі об'єкта проєкту (POM), Maven може керувати складанням, звітом та документацією проєкту [10].

Apache Tomcat - це Open Source реалізація технологій Java Servlet, JavaServer Pages, Java Expression Language та Java WebSocket, Java сервлет, сторінки JavaServer, мова виразів Java та специфікації Java WebSocket [11].

					IA61.260БАК.003 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

JUnit 5 - це бібліотека для модульного тестування, наступне покоління Junit, мета якого - створити сучасну основу для тестування на розробці на стороні JVM. Сюди входить орієнтованість на Java 8 і вище, а також включення багатьох різних стилів тестування.

Scala стала додатковою мовою програмування в проєкті. Scala є молодою мовою програмування, вона була розроблена в 2003, Scala поєднує об'єктно-орієнтоване та функціональне програмування однією стислою мовою високого рівня. Статичні типи Scala допомагають уникнути помилок у складних програмах, а її JVM та JavaScript під час виконання дозволяють створювати високопродуктивні системи з легким доступом до величезних екосистем бібліотек [12].

В дипломному проєкті Scala використовувалася для написання навантажувальних тестів за допомогою фреймворка Gatling.

JavaScript стала другою додатковою мовою програмування - це легковагова мова програмування, яка використовує JIT-компіляцію. Вона прототипно-орієнтована та підтримує декілька парадигм програмування(ООП, функціональне й інші), має динамічну типізацію з підтримкою імперативного та декларативного стилю програмування [13].

Gatling - це дуже потужний інструмент для перевірки навантаження. Він розроблений для зручності використання, пошуку несправностей та високої продуктивності.

Gatling має чудову підтримку протоколу HTTP, що робить його інструментом вибору для тестування завантаження будь-якого HTTP-сервера. Оскільки основний двигун насправді є протокольним агностиком, цілком можливо реалізувати підтримку інших протоколів [14]. Наприклад, Gatling в даний час також постачає підтримку JMS [14].

Наявність сценаріїв, визначених у кодї та ресурсоефективних, є двома основними перевагами Gatling. Спираючись на виразний DSL, сценарії пояснюють себе.

					IA61.260БАК.003 ПЗ	Арк.
						18
Змн.	Арк.	№ докум.	Підпис	Дата		

Вони прості в обслуговуванні і можуть зберігатися у системі управління версіями.

Архітектура Gatling є асинхронною, усюди, де базовий протокол, такий як HTTP, може бути реалізований не блокуючим способом [14]. Така архітектура дозволяє реалізовувати віртуальних користувачів у вигляді повідомлень замість виділених ниток, що робить їх дуже дешевими ресурсами. Таким чином, запуск тисяч одночасних віртуальних користувачів не є проблемою.

Gatling має інформаційну панель показників, яка створюється після виконання тестів [14]. Звіти зберігаються як веб ресурси, які можна зберігати та передавати для аналізу та порівняння показників. Звіт Gatling інтерактивний, через це можна проводити аналіз конкретних запитів.

Gatling має перевагу над деякими фреймворками для навантажувального тестування, так як він використовує набір інструментів Akka, що заснований на акторській моделі, яка є асинхронною. Завдяки Akka, Gatling не створює окремі потоки для кожного юзера. Багатопоточність здійснена за допомогою повідомлень між діючими особами, які дозволяють змінювати список юзерів за допомогою одного потоку.

Gatling краще справляється з навантаження ніж, наприклад, JMeter, через що вдається зберегти процесорний час і оперативну пам'ять, й тому дозволяє імітувати набагато більшу кількість юзерів.

Gatling тести зрозумілі для людини, через те, що Gatling використовує специфічну для домену мову. Це допомагає різним людям працювати над однаковими сценаріями та не витратити час на передачу знань.

Gatling сумісний з різними платформами CI, через те, що його можна виконати з командного рядка.

Модель Publish Subscribe дозволяє асинхронно транслювати повідомлення в різні частини системи, модель зображена на рисунку 4.3 [15].

					ІА61.260БАК.003 ПЗ	Арк.
						19
Змн.	Арк.	№ докум.	Підпис	Дата		

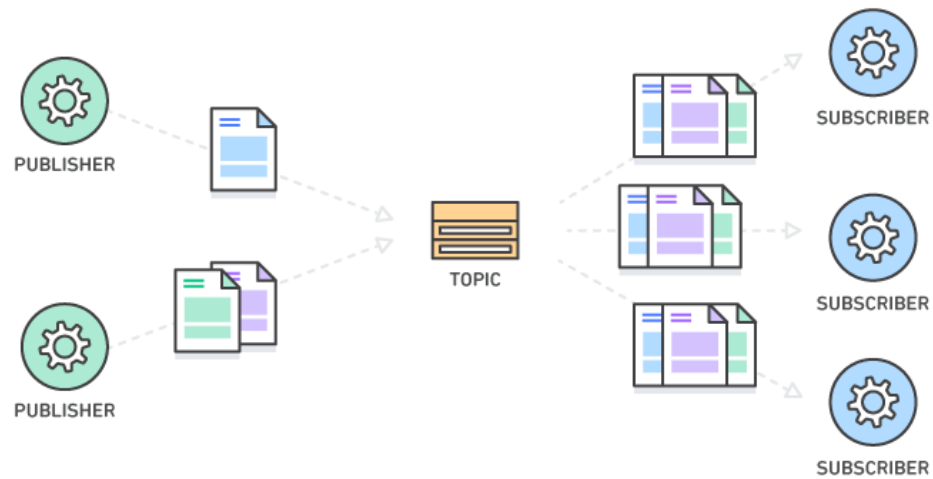


Рисунок 4.3 – Модель архітектури Pub-Sub

Розміщення по черзі повідомлень, тема повідомлень забезпечує легкий механізм трансляції асинхронних сповіщень про події та кінцеві точки, що дозволяють компонентам програмного забезпечення підключатися до теми для того, щоб надсилати та отримувати ці повідомлення [15]. Щоб транслявати повідомлення, компонент, який називається видавцем, просто підштовхує повідомлення до теми [15]. На відміну від черг повідомлень, які створюють пакетні повідомлення до їх завантаження, теми повідомлень передають повідомлення без черги чи з дуже малою чергою, і негайно висилають їх усім підписникам. Усі компоненти, які підписалися на тему, отримуватимуть кожне повідомлення, яке транслюється, за винятком випадків, коли підписник не встановив політику фільтрації повідомлень.

Event-Driven архітектура має такі переваги як асинхронність, малу зв'язність, легку масштабованість, відновлюваність після збоїв.

Асинхронність дозволяє ресурсам вільно переходити до наступного завдання, коли їх одиниця роботи буде завершена, не турбуючись про те, що сталося раніше чи станеться далі. Вони також дозволяють чергувати чи будувати події, що заважає споживачам чинити тиск на виробників чи блокувати їх.

При малій зв'язності сервіси не потребують (і не повинні) знання або залежностей від інших служб. Під час використання подій сервіси

функціонують самотійно, не знаючи інших служб, включаючи дані про їх реалізацію та транспортний протокол. Послуги за моделлю подій можна оновлювати, тестувати та розгортати незалежно та простіше.

Так як кожна частина в цій моделі є незалежною, то вона може масштабуватися окремо від іншої системи. Це додає більшої гнучкості, коли розробник може масштабувати саме ті модулі, які цього потребують.

Підтримка відновлення. Керована подіями архітектура з чергою може відновити втрачену роботу шляхом "повторного відтворення" подій з минулого. Це може бути цінним для запобігання втрати даних, коли споживачеві потрібно відновити.

Висновки до розділу 2

Базуючись на результатах порівняльного аналізу с існуючими рішеннями була розроблена функціональна схема застосунку, яка детально описує кожен окремий процес функціонування застосунку, не вдаючись в деталі технічної реалізації. Кожний процес, та ресурси, які приймають в ньому участь були поверхнево описані.

Для технічної реалізації проєкту була обрана вірна архітектура, яка базувалася на шаблонах проєктування для асинхронних систем. Були обрані раціональні технології, які відповідають основним критеріям – це безкоштовність, підтримуваність, сучасність, швидкодія, надійність. Завдяки слідуванню сучасним стандартам набір технологій є звичним та зрозумілим фахівцям, які працюють з хмарними та Java орієнтованими застосунками. Усі елементи (за виключенням DynamoDB) є безкоштовними, тож замінюючи хмарну DynamoDB локальною версією для розробників, було можливо створити проєкт без додаткових витрат на оплату інших компонентів та розмістити його як Open Source. Завдяки використанню сучасного фреймворка, бібліотека є платформно-незалежною, та легкою для розгортання завдяки вбудованому контейнеру. Правильний вибір технологій дозволив

					ІА61.260БАК.003 ПЗ	Арк.
						21
Змн.	Арк.	№ докум.	Підпис	Дата		

створити потужну систему модульного, інтеграційного та навантажувального тестування, що дозволяє підтримувати належну якість застосунку.

Вибір різних та оптимальних мов програмування для різних задач підвищує якість проєкту, а їх гармонічне поєднання (Scala є дуже схожою на Java) не створює додаткових складнощів для проєкту.

Орієнтованість на AWS застосунки, як на цільову аудиторію користувачів, є влучним вибором, бо AWS є найпопулярнішим хмарним провайдером сьогодення, що забезпечує широке коло потенційних користувачів.

					ІА61.260БАК.003 ПЗ	Арк.
						22
Змн.	Арк.	№ докум.	Підпис	Дата		

3 РЕАЛІЗАЦІЯ БІБЛІОТЕКИ

3.1 Цикл ініціалізації бібліотеки

Процес роботи бібліотеки складається з фаз зображених на рисунку 3.1:

- імпортування старту бібліотеки в конфігурації клієнтського застосунку;
- підняття бінів бібліотеки;
- пошук анотованих класів для створення запитів;
- побудова метадати по знайденим класам;
- завершення підняття контексту й перехід в робочий стан;
- обробка запитів, що надходять від клієнтського застосунку.

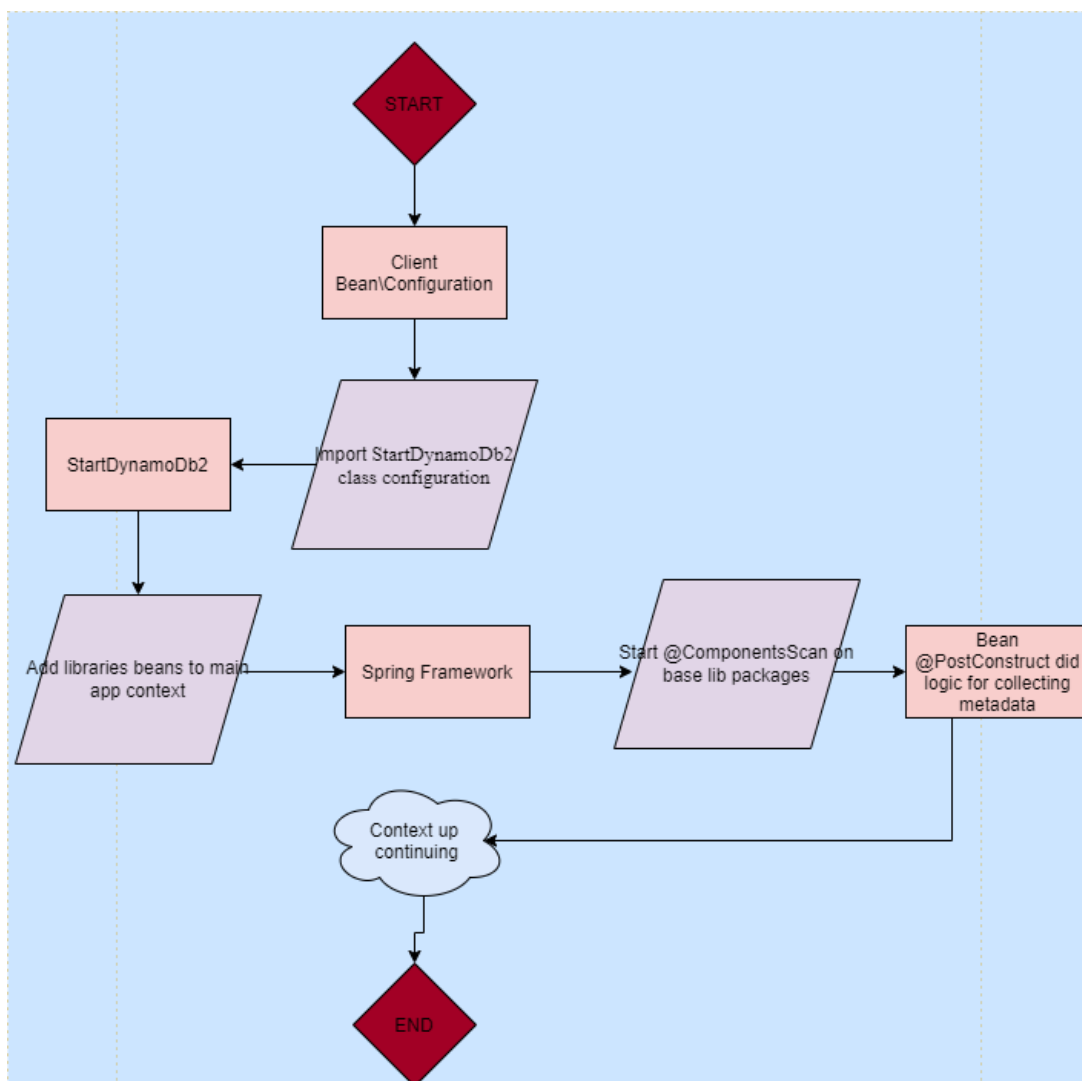


Рисунок 3.1 – Процес інтеграції з бібліотекою

Змн.	Арк.	№ докум.	Підпис	Дата

ДОДАТОК А – лістинг класів розробленої бібліотеки та клієнтського тестового додатку.

Розглядати реалізацію краще почати в такому ж порядку, як буде користуватися бібліотекою клієнт – тобто зі стадії інтеграції.

Перше, що треба зробити в клієнтському застосунку це додати стартер бібліотеки за допомогою конфігурації контексту клієнтського застосунку.

ApplicationContext - це головний інтерфейс в Spring-додатку, який надає інформацію про конфігурацію програми. Він доступний тільки для читання під час виконання, але може бути перезавантажений при необхідності і підтримки додатком.

ApplicationContext надає:

- фабричні методи біна для доступу до компонентів додатка;
- можливість завантажувати файлові ресурси в загальному вигляді;
- можливість публікувати події та реєструвати обробники на них;
- можливість працювати з повідомленнями з підтримкою інтернаціоналізації;
- спадкування від батьківського контексту.

Важливо розуміти, що при виконанні біни бібліотеки додаються до контексту клієнтського застосунку, а при тестування бібліотеки є можливість піднімати її контекст окремо.

В процесі ініціалізації та підняття контексту застосунку разом із бінами бібліотеки піднімаються й біни клієнтського застосунку. Біни можна поділити на такі, що будуть створені при піднятті контексту, й такі, що будуть ініціалізовані при спробі пошуку його по типу через контекст. Біни також поділяються на обов'язкові та необов'язкові.

Spring Framework застосовує Fail-Fast підхід стосовно обов'язкових бінів, це означає, що якщо, хоча б один бін не вдасться ініціалізувати, оновити та додати до контексту – Spring буде вважати весь застосунок неробочим і не запустить його. Це важливо, бо, якщо клієнтський застосунок

					ІА61.260БАК.003 ПЗ	Арк.
						24
Змн.	Арк.	№ докум.	Підпис	Дата		

буде мати невалідні біни, або такі, що конфліктують з бінами бібліотеки – застосунок може не працювати, або працювати не так як це очікується.

Наступним кроком клієнтський застосунок має додати у контекст `DynamoDbAsyncClient` бін, який має всередині налаштування параметрів підключення до бази даних, такі як – `accessKey`, `secretKey` та `region`. Створити цей бін та надати в користування задача саме клієнтського застосунку, для того, щоб бібліотека не залежала від параметрів підключення і не потребувала знань приватних даних – це забезпечує гнучкість системи. Для тестування бібліотека створює власний бін, `DynamoDbAsyncClient`, який створює підключення до локального тестового інстансу `DynamoDB`. В нормальному режимі бібліотека очікує систему, що буде мати зв'язок лише з однією базою даних, тобто мати лише один бін `DynamoDbAsyncClient`, але завдяки гнучкій роботі через шаблон інверсії контролю – ця логіка може бути змінена, клієнт може вказати конкретне ім'я біна, з яким повинна працювати бібліотека, і мати в контексті зв'язок з більш ніж однією базою даних.

Наступним кроком необхідно вказати значення пакету або пакетів, які має просканувати програма для пошуку кандидатів. Spring дозволяє динамічно змінювати значення полів біна за допомогою додаткових конфігураційних ресурсів. Ці ресурси налаштовуються `PropertyPlaceholderConfigurer`, ці налаштування необхідні, адже фреймворк дозволяє дуже широкий вибір того, як конфігурувати свій додаток, це можуть бути не лише файли, а й окремі програми або процеси, зовнішні потоки, тощо. Наприклад, в застосунках орієнтованих на хмару, гарною практикою є зберігання конфігурацій окремо від коду в спеціальному місці, а зберігання приватних налаштувань, в окремому захищеному місці.

Додаток користувача, в своїх додаткових конфігураціях, чим би вони не були, вказує конфігурацію з іменем “spring.data.dynamodb.sdkv2.packages.entities” зі значенням імені пакету або пакетів, в якому знаходяться його об'єкти бази даних.

					ІА61.260БАК.003 ПЗ	Арк.
						25
Змн.	Арк.	№ докум.	Підпис	Дата		

Далі по класам, що знаходяться в цьому пакеті потрібно виконати пошук кандидатів. Існує декілька варіантів реалізувати таку логіку. Якщо застосунок не використовує фреймворків, це треба робити через ClassLoader. Такий алгоритм має багато складнощів.

По-перше, будь-який Java застосунок має декілька Class-Loader для завантаження різних типів даних(ресурси, статичні класи, контекстні класи). Через цей факт виникають проблеми того, що бібліотека буде або накладати обмеження на формат класів для клієнтських entity, або буде вимушена завжди слухати декілька завантажувачів.

По-друге, в Java існує можливість зробити так званий Hot-Swap класів(заміна класів без зупинки застосунку) або завантажити додаткові ресурси, то ж при цьому підході буде недостатньо прослухати всі завантажені класи лише при старті.

По-третє, в базовій реалізації, немає можливості оперувати лише з класами в певному пакеті – це означає, що потрібно буде провести аналітику всіх завантажених класів. Навіть у малих додатках це буде значна кількість через використання залежностей.

Вибір цього варіанту змусив би додати до бібліотеки зайвої неоптимальної логіки, яка б робила її повільнішою, призводила до помилок.

В проєкті ця логіка реалізована більш оптимально завдяки перевизначенню алгоритму пошуку бін-кандидатів Spring Framework. Для того, щоб виконувати свій функціонал – Spring змушений прослуховувати всі ClassLoader-и застосунку і робить це оптимально. Spring фільтрує ресурси по анотаціям фреймворку, таким як @Bean, @Component, @Service, @Repository, та іншим, знаходячи клас з відповідною анотацією – Spring вважає його бін кандидатом.

Алгоритм реалізований в бібліотеці полягає у реалізації шаблону Декоратор, який розширює можливості системи Spring – фактично виконується додавання анотації @Table, і якщо клас її має – він додається у список кандидатів, які алгоритмом додаються в окремий Set класів.

					ІА61.260БАК.003 ПЗ	Арк.
						26
Змн.	Арк.	№ докум.	Підпис	Дата		

Крім того, Spring надає можливість оптимізації алгоритму, завдяки якому пошук можна проводити тільки по окремому пакету. В бібліотеці, звісно, виконана ця оптимізація, клієнтський застосунок надає ім'я пакету для пошуку.

Останнім кроком ініціалізації створюється метадата(Об'єкт класу Item) на базі якої будуть будуватися запити до БД.

Spring дозволяє додавати спеціалізовану логіку ініціалізації бінів за допомогою анотації `@PostConstruct` – вона дозволяє виконати додаткову логіку, після виконання базового конструктору.

Розгляньмо діаграму класів бібліотеки [IA61.260БАК.008 Д3]. При додаванні `StartDynamoDb2` активується `@ComponentScan` по базовим пакетам бібліотеки – `repository` та `service`, в результаті чого будуть створені біни типів `DynamoDbCrudRepo` та `TargetComponentsLoaderService`. В реалізації `TargetComponentsLoaderService` метод `init` анатовано `@PostConstruct`, це означає, що Spring Framework має виконати вказану додаткову логіку. Приклад використання в проєкті зображено на рисунку 3.2.

```
36      @PostConstruct
37      public void init() {
38          createClassesSet();
39          createItemList();
40      }
```

Рисунок 3.2 – Додаткова ініціалізація біна

З цієї ініціалізації фактично починається старт функціональності бібліотеки. Задача полягає в отриманні усіх класів, що є кандидатами на створення метаданих для них та генерації запитів.

В процесі створення метадати по сету (колекція даних в Java) класів, запускається ітерація, котра бере кожний клас-кандидат та намагається проаналізувати зважаючи на можливість наявності помилок в описанні класу.

Першою дією алгоритм отримує усі поля класу за допомогою Java Reflections. Використання Java Reflections – це повільна операція, і взагалі до використання цього пакету Java слід ставитись обережно, адже він здатний обходити захисні механізми ООП, такі як інкапсуляція, і втручатися у життєвий цикл сторонніх об’єктів. Важливо розуміти, що дію діставання полів алгоритм виконає лише раз і на етапі запуску ПЗ, а не під час роботи, тому це не створить додаткових проблем швидкодії програми.

Другою дією алгоритм викликає метод “defineTableName” з пакету “com.tiurinalery.springdata.sdk2.parser”, цей метод перевіряє наявність, та якщо є – дістає ім’я таблиці відповідне цьому класу с параметру анотації. Таким чином створюється відношення один до одного між класом та таблицею.

Третьою дією алгоритм викликає метод “defineKeyAndAttribute” з того ж пакету, його задача опрацювати ключі та атрибути. Якщо аргумент є ключем, то з’ясовується його назва в схемі, та тип ключа в AWS DynamoDB – він може бути Hash або Range, ці данні вказуються користувачем. Якщо, елемент є просто атрибутом, то береться лише його ім’я в схемі документа.

Таким чином формується один об’єкт класу Item, на кожний правильно описаний клас-кандидат.

Після завершення підняття та оновлення контексту клієнтського застосунку – програма переходить в стан RUNNING та може опрацьовувати запити й генерувати запити до БД.

3.2 Опис процесу генерації запитів в застосунку

Після переходу застосунку в стан RUNNING, бібліотека може починати генерувати запити. Взаємодія з бібліотекою наразі відбувається через інтерфейс DynamoDbCrudRepo, згодом при збільшенні функціоналу будуть додані й додаткові інтерфейси для підтримки рідковживаних операцій.

					ІА61.260БАК.003 ПЗ	Арк.
						28
Змн.	Арк.	№ докум.	Підпис	Дата		

У клієнта є два шляхи використання – зробити впровадження базового інтерфейсу або унаслідуватись від нього, та перевизначити його, чи додати свою логіку.

Контракт `DynamoDbCrudRepo` розроблений таким чином, щоб підтримувати CRUD(Create, Read, Update, Delete) операції без зайвих складнощey, саме тому клієнту в більшості ситуацій підійде базовий інтерфейс.

Для того, щоб створити новий запис або оновити старий – все що потрібно зробити клієнту це викликати метод `save(Object)` в інтерфейсі й передати бажаний об'єкт як параметр. В свою чергу бібліотека перевірить якого класу цей об'єкт, і спробує знайти метадату(об'єкт класу `Item`) для цього класу, якщо вона буде знайдена – бібліотека отримає з неї відомості про структуру таблиці та зв'язок між полями об'єкта та полями в БД, а також які атрибути являються ключами. На базі цих відомостей бібліотека підставить у згенерований “скелетон” запиту данні з об'єкту, який прислав клієнтський застосунок і виконає запит. Як результат буде повернута обгортка для асинхронної роботи `CompletableFuture` над відповіддю з БД.

Для того, щоб отримати запис в користувача є декілька можливостей, які надає `DynamoDB` й відповідно підтримує бібліотека. Запис можна отримати по головному ключу, використавши метод `findById` й передавши ключ як параметр, або через вторинний індекс(будь яке інше поле, яке проіндексоване) викликавши метод `findBySecondaryIndexes` передавши заповненими значення проіндексованих полів як атрибут. Режим роботи бібліотеки буде еквівалентним до створення, вона знайде метадату для цього класу, побудує “скелетон” запиту та заповнить його даними базуючись на запиті та наданих атрибутах від користувача. Результатом роботи також буде обгортка `CompletableFuture` над результатом з БД, що дозволить клієнтському застосунку додати асинхронну додаткову логіку.

					ІА61.260БАК.003 ПЗ	Арк.
						29
Змн.	Арк.	№ докум.	Підпис	Дата		

Для користувача робота з БД перетворюється на модель виклика методу й отримання результату, усі маніпуляції над створенням запиту бере на себе створена бібліотека.

3.3 Опис використаних шаблонів проєктування в архітектурі

Декоратор – це структурний шаблон, який використовують для надання додаткової логіки існуючим об’єктам, схема якого зображена на рисунку 3.3 [16, 17].

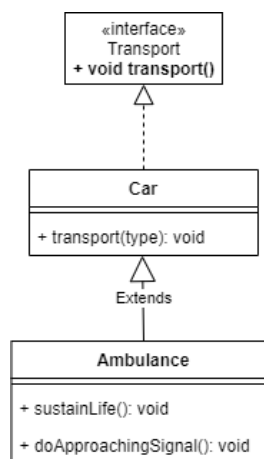


Рисунок 3.3 – Схема шаблону декоратор

Інтерфейс Transport, має єдиний метод, який відповідає за пересування, й клас Car здійснює цей інтерфейс, тобто всі машини вміють їздити. Але з’являється новий тип машин – швидка допомога, яка теж вміє їздити, але й ще підтримувати життя хворого та сигналізувати. Тож, щоб не ускладнювати клас Car, та слідувати ООП принципам, клас Ambulance наслідує його та розширює. Цей процес розширення функціональності і є реалізацією шаблону.

Будівельник – це структурний шаблон, мета якого відокремити логіку створення об’єкту від логіки представлення об’єкта, схема якого зображена на рисунку 3.4 [16,17].

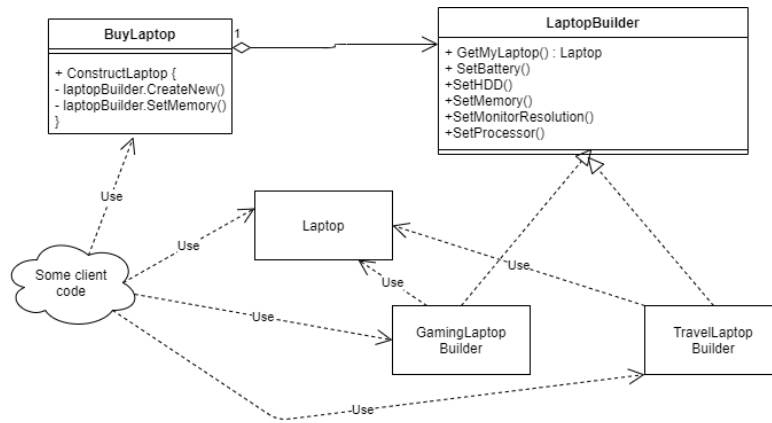


Рисунок 3.4 – Схема шаблону будівельник

Цей дизайн шаблон є корисним, адже створення нових об'єктів часто стає стандартизованим для певних задач. Наприклад, для комп'ютерів – ноутбук для ігор має складатися з одних обов'язкових характеристик, а з іншої сторони для ноутбуку для бізнесу – важливі зовсім інші.

Інверсія контролю – шаблон, який оснований у відповідності до принципу “єдиного обов'язку”, при його реалізації об'єкт передає процес “впровадження” зовнішньої залежності іншому компоненту, який ще називають впровадником [5,17]. При цьому підході об'єкти відповідають лише за свою єдину задачу, а про їх створення, контролювання життєвого циклу, та надавання у користування піклується впровадник.

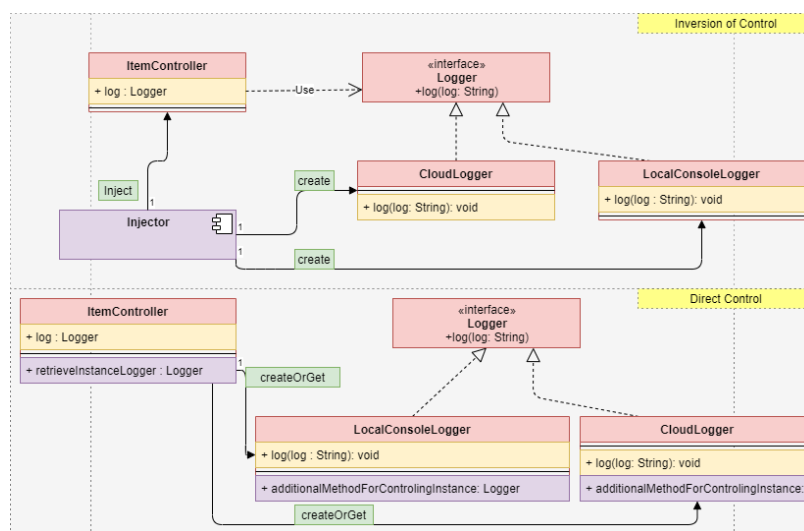


Рисунок 3.5 – Порівняльна схема прямого та інверсійного контролю

Дивлячись на рисунок 3.5, прямий контроль може здаватися значно кращим рішенням, адже ідея створення нового об'ємного компоненту часто здається нераціональною. Однак, при відсутності впровадника, його функції розподіляються між всіма об'єктами системи, що ускладнює її та підвищує зв'язність компонентів. При зростанні кількості об'єктів в системі - кількість дубльованого коду стає все більшою, і потреба у впроваднику стає очевидною

Висновки до розділу 3

Була розроблена бібліотека для автоматичної генерації найчастіше вживаних запитів.

Першим кроком створена технічна діаграма, яка враховувала:

- процес ініціалізації бібліотеки;
- процес інтеграції бібліотеки с користувацьким ПЗ;
- робочий режим бібліотеки;
- обробку помилок бібліотеки;
- можливість автоматизованого тестування бібліотеки.

Була розроблена низькорівнева архітектура застосунку, яка базувалася на шаблонах проєктування рекомендованих для Spring застосунків.

Були влучно перевикористані можливості Spring Framework для досягання цілей проєкту, а саме для роботи з завантажуваними класами, інверсією контролю та інших.

Був розроблений простий інтерфейс взаємодії з бібліотекою для користувачів, який дає можливість розробникам одразу використовувати бібліотеку, а не вивчати її внутрішню будову.

Була закладена можливість додавання нових інтерфейсів взаємодії для подальшого розширення функціоналу.

Були створені функціональні тести бібліотеки та автоматизована система зборки та розгортання проєкту.

					ІА61.260БАК.003 ПЗ	Арк.
						32
Змн.	Арк.	№ докум.	Підпис	Дата		

4 СТВОРЕННЯ ТЕСТОВОГО ДОДАТКУ З МЕТОЮ ТЕСТУВАННЯ БІБЛІОТЕКИ

4.1 Опис принципів побудови схем нереляційних баз даних.

Побудова схеми нереляційної бази даних вкрай відрізняється від побудови реляційних моделей. Це зумовлено перш за все сферою використання цих баз даних – одним з головних випадків, коли треба використовувати нереляційну базу – невизначеність схеми та зав'язків, часті зміни, тощо. Зазвичай, схема одного документу(аналог таблиці в реляційних базах) проєктується так, щоб мати всі данні для одного повного бізнес процесу.

Наприклад, якщо треба створити базу даних автомобілів, магазинів та покупців – буде створено 3 або менше таблиць на відміну від реляційної моделі, де розробник буде піклуватися про нормалізацію бази. Нереляційні бази даних мають великий потенціал в додатках, які мають велику кількість записів, та порівняно нечасті та нереляційні запити – це властиво системам моніторингу, кешу, відеоіграм, IoT системам.

Зазвичай, кожний документ має так званий HashKey, який виступає аналогом Primary Key в реляційних базах даних, по якому виконується пошук. Якщо пошук необхідно проводити більш ніж по одному полю, то поля які не входять у склад ключа – індексують, це значно пришвидшує пошук по цьому полю(який насправді вже відбувається не по полю, а по індексу) ціною збільшення об'єму даних, які потрібно зберігати.

Для того, щоб будувати в нереляційних базах композитні ключі, такі що складаються з більш ніж одного поля – необхідно використовувати RangeKey.

HashKey в нереляційному сховищі – може бути лише 1 полем в записі, по якому вираховується хеш-код. Часто бізнесу необхідно мати композитні ключі з 2 або більше полів, які будуть характеризувати певний унікальний об'єкт.

					ІА61.260БАК.003 ПЗ	Арк.
						33
Змн.	Арк.	№ докум.	Підпис	Дата		

Наприклад, існує компанія, що продає різні сервіси користувачам по підписці, важливе обмеження – у кожного користувача може бути лише 1 підписка на кожен сервіс.

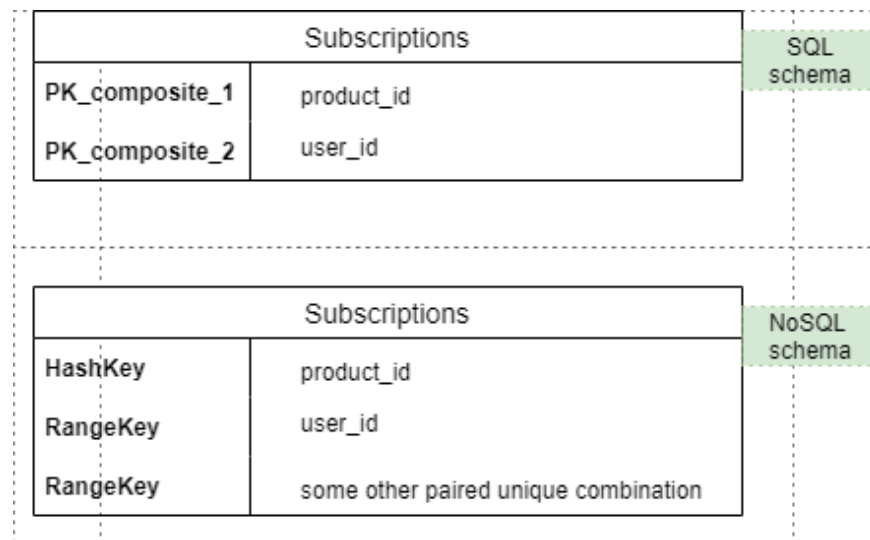


Рисунок 4.1 – Порівняння схем SQL та NoSQL ідентифікаторів запису

В такому випадку ідентифікатор сервісу, який був названий `product_id` стане `HashKey`, а `user_id` – відповідно `RangeKey`, як зображено на рисунку 4.1. Для того, щоб така схема була вірною, треба суворо підтримувати унікальність пари, в більшості NoSQL СУБД це робить сама база даних.

Нереляційні бази можуть включати більш ніж один `RangeKey`, це необхідно, бо одна таблиця описує цілий бізнес процес. Кожна пара `Hash` і `Range` ключів буде проіндексована невпорядкованим індексом. Якщо всі майбутні записи мають бути впорядкованими, то базову поведінку слід перевизначити і використовувати впорядкований індекс.

DynamoDB – це нереляційне сховище типу ключ-значення.

Для того, щоб зробити DynamoDB настільки швидкою, якою вона є, розробникам довелося піти як на якісні алгоритмічні так і на горизонтальні дії задля ефективного масштабування.

З алгоритмічного боку – виклик полягав у тому, що нереляційна база, яка є оптимальною на запис та неоптимальною на читання, мала стати

оптимальною по швидкості для обох операцій та ще й на величезних об'ємах даних.

Головною оптимізацією є розподілене зберігання посилань на дані в оперативній пам'яті, та реальних даних в загальній пам'яті.

Безпосередньо дані зберігаються на SSD носіях, які автоматично створюють репліки в різних зонах доступу(кожний регіон в хмарних застосунках поділений на зони доступу). Кожна партіція зберігає не більш ніж 10 Гб даних – це в середньому 25 млн записів.

Кількість партіцій обраховується за формулою:

$$(\text{readCapacityUnits} / 3000) + (\text{writeCapacityUnits} / 1000), \quad (4.1)$$

де `readCapacityUnits` та `writeCapacityUnits` – значення, які обирає адміністратор AWS інфраструктури для кожної таблиці.

Ці значення впливають на швидкодію та ціну.

4.2 Огляд можливих середовищ розгортання DynamoDB

DynamoDB можна розвернути трьома наступними способами – в одиночному докер контейнері, в Kubernetes кластері контейнерів та безпосередньо в хмарному середовищі. Далі більш детально описаний кожний з варіантів розгортання.

Хмарне середовище, безумовно, з функціональної точки зору це – найкраще рішення, але ж воно й найдорожче. Для того, щоб проводити тестування, необхідно було б придбати не лише місце для DynamoDB, а й контейнер EC2, які б знаходилися в одній VPC, ІО якої теж необхідно було б оплатити. Оренда лише DynamoDB таблиці неефективна, бо відстань між локальним комп'ютером тестувальника, де буде розгорнуто застосунок, та сервером буде значною, тому перевірка швидкодії буде впертися не в алгоритмічну складність а в швидкодію та стабільність інтернет з'єднання.

					ІА61.260БАК.003 ПЗ	Арк.
						35
Змн.	Арк.	№ докум.	Підпис	Дата		



Рисунок 4.2 – Карта Зон Доступності в Європі

На рисунках 4.2 та 4.3 можна побачити, що найближчий з наявних серверів знаходиться у Франкфурті.

eu-central-1	Europe (Frankfurt)	Not required	No
eu-west-1	Europe (Ireland)	Not required	No
eu-west-2	Europe (London)	Not required	No
eu-west-3	Europe (Paris)	Not required	No
eu-north-1	Europe (Stockholm)	Not required	No

Рисунок 4.3 – Міста розташування серверів AWS в Європі

В середньому пінг до Франкфурта складає 46 мс, як показано на рисунку 4.4, що порівняно зі швидкістю роботи DynamoDB 1-9 мс – дуже багато, тому модель тестування “Локальний ПК – DynamoDB в хмарі” не мала б ніякого сенсу.

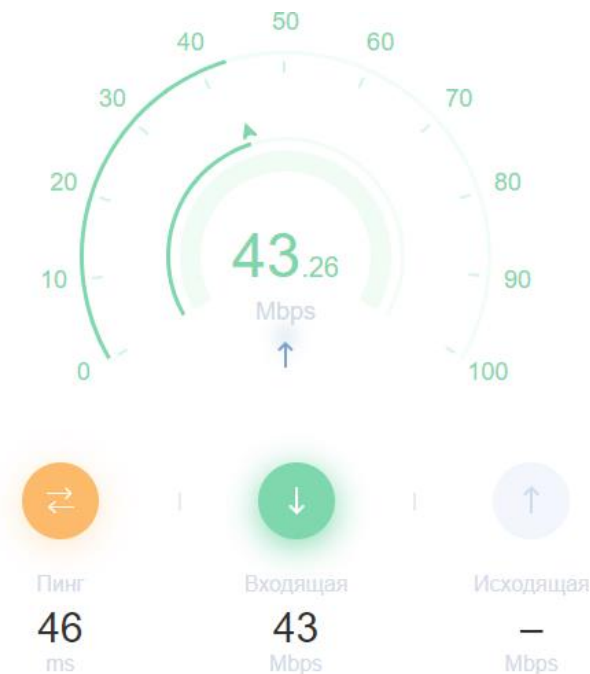


Рисунок 4.4 – Показники швидкості інтернет з'єднання до сервера

Проблему інтернет з'єднання можливо вирішити шляхом оренди кластеру мало-потужних контейнерів в хмарі та об'єднання їх з DynamoDB в 1 VPC. На рисунках 4.5, 4.6 та 4.7 зображені ціни на DynamoDB, VPC та EC2.

Region: Europe (Frankfurt) ▾		
Monthly Commitment	Upfront: 1-Year	Hourly: 1-Year
100 Write Capacity Units	\$183.00	\$0.0157
100 Read Capacity Units	\$36.60	\$0.0032

Рисунок 4.5 – Ціна однієї таблиці DynamoDB

Region: Europe (Frankfurt) ▾	
Price per NAT gateway (\$/hour)	Price per GB data processed (\$)
\$0.052	\$0.052

Рисунок 4.6 – ціна передачі 1 Гб по VPC

Локальний ПК здатний імітувати близько 300 DynamoDB units та по потужності еквівалентний обраному кластеру, тож хоча б еквівалентне тестове хмарне середовище коштувало б 207,55\$ в місяць:

$$\text{DynamoDB_Price} = (3 \times \text{WCU_HOURLY} + 3 \times \text{RCU_HOURLY}) \times \text{DAY_IN_MONTH} \times \text{NUMBER_TESTS_RUNNING_PER_DAY} \times \text{TIME_1_CYCLE_RUNNING}, \quad (4.2)$$

де DynamoDB_Price – ціна Dynamo БД;

WCU_HOURLY – погодинний об'єм абстрактних одиниць для запису;

RCU_HOURLY – погодинний об'єм абстрактних одиниць для зчитування;

DAY_IN_MONTH – днів у місяці;

NUMBER_TESTS_RUNNING_PER_DAY – кількість тестів, що проходять за день;

TIME_1_CYCLE_RUNNING – час проходження тестів.

WCU_HOURLY = 0.0157;

RCU_HOURLY = 0.0032;

DAY_IN_MONTH = 30;

NUMBER_TESTS_RUNNING_PER_DAY >= 10;

TIME_1_CYCLE_RUNNING = 0.45 hour;

DynamoDB_Price = 7.65 \$ за місяць [18].

$$\begin{aligned} \text{VPC_I/O_Price} = & \text{SIZE_OF_1_QUERY} \times 300 \times 3600 \times \\ & \text{TIME_1_CYCLE_RUNNING} \times \\ & \text{NUMBER_TESTS_RUNNING_PER_DAY} \times \text{DAY_IN_MONTH} \times 0.052 \\ & \times 2, \end{aligned} \quad (4.3)$$

де VPC_I/O_Price – віртуальна приватна мережа;
 SIZE_OF_1_QUERY – розмір одного запиту;
 300 – запитів в секунду;
 3600 – перехід від секунд до годин;
 TIME_1_CYCLE_RUNNING – час проходження тестів;
 NUMBER_TESTS_RUNNING_PER_DAY – кількість тестів, що проходять за день;
 DAY_IN_MONTH – днів у місяці;
 SIZE_OF_1_QUERY ~ 10Kb = 1×10^{-5} Gb ;
 VPC_I/O_Price = 151.63 \$ за місяць [19].

$$\begin{aligned} \text{TOTAL_PRICE} &= \text{EC2_Price} + \text{DynamoDB_Price} \\ &+ \text{VPC_I/O_Price} = 207,55 \$, \end{aligned} \quad (4.4)$$

де EC2_Price – ціна обчислювальної потужності в хмарі [18].

Amazon Elastic Block Storage (EBS) [Info](#)
 Attach persistent block storage volumes for your Amazon EC2 Instances.

Storage per instance

Storage for each EC2 instance
 Provisioned IOPS SSD (io1)

Provisioning IOPS
 3

Storage amount
 30 GB

► Show calculations

Amazon EC2 estimate

Amazon EC2 Instance Savings Plans Instances (monthly)	34.16 USD
Amazon Elastic Block Storage (EBS) pricing (monthly)	14.11 USD
Total monthly cost:	48.27 USD

Cancel [Add to my estimate](#)

Рисунок 4.7 – місячна ціна EC2 оренди

Для того, щоб створити хоча б еквівалентне до локального середовища хмарне середовище, необхідно було б витратити 207,55 \$ в місяць. Єдиною

					ІА61.260БАК.003 ПЗ	Арк.
						39
Змн.	Арк.	№ докум.	Підпис	Дата		

причиною не створювати повноцінне тестове середовище в хмарі є фактична відсутність коштів на його підтримку.

Розгортання додатку в Kubernetes кластері локально розглядалося як можливий варіант тестування, який був протестований та відхилений. Головною проблемою локального навантажувального тестування є внутрішня пропускна здатність передачі даних та розмір операційної пам'яті. Розбивання на окремі контейнери та створення кластеру тільки погіршувало ситуацію з ефективністю локального тестування.

Найкращим варіантом було розгортання бази даних DynamoDB в одиночному контейнері – це рішення було простим, дешевим з точки зору оперативної пам'яті, не перевантаженим з точки зору пропускної здатності та найголовніше – повністю безкоштовним.

4.3 Створення тестової бази даних

Для того, щоб підняти вже налаштований образ Linux зі встановленою DynamoDB необхідно завантажити з центрального Docker репозиторію образ dynamodb-local. Після завантаження, можна піднімати контейнери з нею. Оскільки в процесі розробки, проводити тестування треба часто – необхідно автоматизувати та зберегти процес налаштування. Docker надає інструментарій для цього – за допомогою так званих Dockerfile та docker-compose файлів.

За допомогою цих інструментів можна один раз задати сценарій розгортання та налаштування, й далі піднімати готовий образ однією командою без жодних подальших налаштувань. Такі опції корисні, як при тестуванні вручну, так і при використанні CI/CD систем.

Конфігураційний скрипт – ДОДАТОК Б.

Для запуску необхідно виконати просту команду “docker-compose up” і буде отримано тестову БД.

					ІА61.260БАК.003 ПЗ	Арк.
						40
Змн.	Арк.	№ докум.	Підпис	Дата		

Для того, щоб створити тестову таблицю в БД необхідно виконати скрипт з рисунку 4.8 в вбудованій консолі, за допомогою JavaScript SDK DynamoDB.

```
1 var params = {
2   TableName: 'PROD_USER',
3   KeySchema: [
4     {
5       AttributeName: 'UUID',
6       KeyType: 'HASH'
7     }
8   ],
9   AttributeDefinitions: [
10    {
11      AttributeName: 'UUID',
12      AttributeType: 'S'
13    },
14    {
15      AttributeName: 'USERNAME',
16      AttributeType: 'S'
17    }
18  ],
19  ProvisionedThroughput: {
20    ReadCapacityUnits: 300,
21    WriteCapacityUnits: 300
22  },
23  GlobalSecondaryIndexes: [{
24    IndexName: 'idndx_username',
25    KeySchema: [
26      {
27        AttributeName: 'USERNAME',
28        KeyType: 'HASH'
29      }
30    ],
31    Projection: {
32      ProjectionType: 'ALL'
33    },
34    ProvisionedThroughput: {
35      ReadCapacityUnits: 300,
36      WriteCapacityUnits: 300
37    }
38  }]
39 }
40
41 dynamodb.createTable(params, function (err, data) {
42   if (err) ppJson(err); // an error occurred
43   else ppJson(data); // successful response
44 }
45 );
```

Рисунок 4.8 – Скрипт для створення таблиці через консоль

В скрипті необхідно описати схему атрибутів, вказати ім'я таблиці, описати схему ключів, схему додаткових індексів на розмір ресурсів на зчитування\запис.

Для прикладу атрибутів обрані UUID, типу строки, який буде позначений як HashKey, введено додатковий атрибут USERNAME також типу строки, ресурси на зчитування та запис підібрані експериментальним шляхом на стадії навантажувального тестування, починаючи з 300.

```

    "TableDescription" {
      "AttributeDefinitions" [
        0: {
          "AttributeName": "UUID"
          "AttributeType": "S"
        }
        1: {
          "AttributeName": "USERNAME"
          "AttributeType": "S"
        }
      ]
      "TableName": "PROD_USER"
      "KeySchema" [
        0: {
          "AttributeName": "UUID"
          "KeyType": "HASH"
        }
      ]
      "TableStatus": "ACTIVE"
      "CreationDateTime": "2020-05-08T20:07:53.090Z"
      "ProvisionedThroughput" {
        "LastIncreaseDateTime": "1970-01-01T00:00:00.000Z"
        "LastDecreaseDateTime": "1970-01-01T00:00:00.000Z"
        "NumberOfDecreasesToday": 0
        "ReadCapacityUnits": 300
        "WriteCapacityUnits": 300
      }
      "TableSizeBytes": 0
      "ItemCount": 0
      "TableArn": "arn:aws:dynamodb:ddblocal:000000000000:table/PROD_USER"
      "GlobalSecondaryIndexes" [
        0: {
          "IndexName": "idndx_username"
          "KeySchema" [
            0: {
              "AttributeName": "USERNAME"
              "KeyType": "HASH"
            }
          ]
          "Projection" {
            "ProjectionType": "ALL"
            "IndexStatus": "ACTIVE"
          }
          "ProvisionedThroughput" {
            "ReadCapacityUnits": 300
            "WriteCapacityUnits": 300
          }
          "IndexSizeBytes": 0
          "ItemCount": 0
          "IndexArn": "arn:aws:dynamodb:ddblocal:000000000000:table/PROD_USER/index/idndx_username"
        }
      ]
    }
  }

```

Рисунок 4.9 – Результат створення

На рисунку 4.9 є два дуже важливих атрибута.

TableArn – це унікальний ідентифікатор ресурсу в AWS інфраструктурі, його використовують для інтеграції з іншими ресурсами як ID ресурсу.

TableStatus – статус таблиці, цей параметр не дуже важливий, якщо використовувати DynamoDB вручну зі швидкістю людини, однак, якщо таблиці створюються в операційній пам'яті командами з іншої програми, для великих таблиць, для яких одразу необхідно виконати партіціювання – час створення таблиці може бути значним. Важливо розуміти, що працювати з даними таблиці можна починати, коли таблиця перейшла в статус ACTIVE.

					ІА61.260БАК.003 ПЗ	Арк.
						42
Змн.	Арк.	№ докум.	Підпис	Дата		

Щоб перевірити, що над об'єктом DynamoDB були виконані певні дії, можна виконати команду з рисунку 4.10 програмно (використовуючи SDK для підтримуваних мов програмування).

```
1 var params = {
2   TableName: 'PROD_USER',
3 };
4 dynamodb.describeTable(params, function(err, data) {
5   if (err) ppJson(err); // an error occurred
6   else ppJson(data); // successful response
7 });
```

Рисунок 4.10 – Перевірка стану таблиці

Результатом виконання буде поточний стан ресурсу, зображений на рисунку 4.11.

```
{
  "Table": {
    "AttributeDefinitions": [
      {
        "AttributeName": "UUID",
        "AttributeType": "S"
      },
      {
        "AttributeName": "USERNAME",
        "AttributeType": "S"
      }
    ],
    "TableName": "PROD_USER",
    "KeySchema": [
      {
        "AttributeName": "UUID",
        "KeyType": "HASH"
      }
    ],
    "TableStatus": "ACTIVE",
    "CreationDateTime": "2020-05-08T20:07:53.090Z",
    "ProvisionedThroughput": {
      "LastIncreaseDateTime": "1970-01-01T00:00:00.000Z",
      "LastDecreaseDateTime": "1970-01-01T00:00:00.000Z",
      "NumberOfDecreasesToday": 0,
      "ReadCapacityUnits": 300,
      "WriteCapacityUnits": 300
    },
    "TableSizeBytes": 0,
    "ItemCount": 0,
    "TableArn": "arn:aws:dynamodb:ddblocal:000000000000:table/PROD_USER",
    "GlobalSecondaryIndexes": [
      {
        "IndexName": "idndx_username",
        "KeySchema": [
          {
            "AttributeName": "USERNAME",
            "KeyType": "HASH"
          }
        ],
        "Projection": {
          "ProjectionType": "ALL"
        },
        "IndexStatus": "ACTIVE",
        "ProvisionedThroughput": {
          "ReadCapacityUnits": 300,
          "WriteCapacityUnits": 300
        },
        "IndexSizeBytes": 0,
        "ItemCount": 0,
        "IndexArn": "arn:aws:dynamodb:ddblocal:000000000000:table/PROD_USER/index/idndx_username"
      }
    ]
  }
}
```

Рисунок 4.11 – Результат перевірки поточного стану ресурсу

Для того, щоб перевірити, те що база даних, СДК та інші зовнішні інтеграції працюють вірно, на першому етапі треба провести мануальне

					ІА61.260БАК.003 ПЗ	Арк.
						43
Змн.	Арк.	№ докум.	Підпис	Дата		

функціональне тестування, тобто додати та отримати запит через консоль, на подальших етапах – це повинні робити авто-тести.

Для додавання елемента необхідно виконати запит зображений на рисунку 4.12 - вказуємо ім'я таблиці, значення атрибутів, при чому наявність значення для ключа UUID – є обов'язковою, також додаємо параметр ReturnConsumedCapacity, щоб подивитися скільки ресурсів займає один найпростіший запит.

```
var params = {
  TableName: 'PROD_USER',
  Item: { // a map of attribute name to AttributeValue

    UUID: 'uuid-12',
    USERNAME: 'username-15y1'
  },
  ReturnConsumedCapacity: 'TOTAL', // optional (NONE | TOTAL | INDEXES)
};
docClient.put(params, function(err, data) {
  if (err) ppJson(err); // an error occurred
  else ppJson(data); // successful response
});
```

Рисунок 4.12 – Додавання елемента до таблиці

```
{
  "ConsumedCapacity": {
    "TableName": "PROD_USER"
    "CapacityUnits": 2
  }
}
```

Рисунок 4.13 – Результат додавання елемента

На рисунку 4.13 показано, що на виконання найпростішого запиту потрібно було витратити 2 capacity (ємності), це означає, що загальна вага запиту не перевищує $2 \cdot 4\text{Кб}$, бо $1 \text{ capacity} = 4\text{Кб}$ з округленням в більшу сторону, тобто $N \cdot 4\text{Кб} + 1\text{bit}$ це вже $N+1 \text{ capacity}$, де N – це кількість capacity по 4Кб .


```

1 var params = {
2   TableName: 'PROD_USER',
3   Key: {
4     UUID: 'uuid-12'
5   },
6
7   ConsistentRead: true,
8   ReturnConsumedCapacity: 'TOTAL'
9 };
10 docClient.get(params, function(err, data) {
11   if (err) ppJson(err); // an error occurred
12   else ppJson(data); // successful response
13 });

```

Рисунок 4.14 – Отримання елемента

На рисунку 4.15 можна побачити, що діставання цього об'єкту зайняло теж 1 capacity.

```

{
  "Item": {
    "UUID": "uuid-12"
    "USERNAME": "username-15yl"
  },
  "ConsumedCapacity": {
    "TableName": "PROD_USER"
    "CapacityUnits": 1
  }
}

```

Рисунок 4.15 – Отриманий елемент з БД

На рисунку 4.16 зображено запит, який видалив тестову таблицю.

```

1 var params = {
2   TableName: 'PROD_USER',
3 };
4 dynamodb.deleteTable(params, function(err, data) {
5   if (err) ppJson(err); // an error occurred
6   else ppJson(data); // successful response
7 });

```

Рисунок 4.16 – Видалення таблиці

4.4 Створення тестового додатку

Розробка тестового додатку виконує дві основні функції в проєкті – тестування кінцевого результату та наглядна інструкція до використання для клієнтів.

Змн.	Арк.	№ докум.	Підпис	Дата

Діаграма класів тестового застосунку [IA61.260БАК.009 Д4].

Тестовий додаток – це застосунок розроблений на базі фреймворку Spring, відповідно до цільової аудиторії клієнтів. Він має 3 рівні – Controller, Service, Model, важливо зупинити увагу на тому, що відсутній DAO рівень – властивий багатьом Java застосункам, які беруться самі працювати з запитами в БД, в тестовому застосунку цю роль виконує розроблена бібліотека.

Схема ініціалізації тестового застосунку еквівалентна до схеми бібліотеки, клас Application знаходиться в базовому пакеті – імпортує конфігурацію бібліотеки, та починає налаштування контексту тестового застосунку з усіх дочірніх пакетів, як зображено на діаграмі.

Важливо загострити увагу на конфігураційному файлі AbstractDynamoDBConfig та методі dynamoDbAsyncClient(), цей метод створює бін, який забезпечує зв'язок з БД, цей бін згадувався в п.6.1, що саме клієнт має його створити, базуючись на біні AwsCredentialsProvider – цей бін створюється лише за допомогою знань access та secret ключів від БД. Схожий за назвою, dynamoDbClient – це також клієнт AWS, але побудований на блокуючих операціях, він еквівалентний, з функціональної точки зору, до асинхронного та буде використовуватись для порівняння.

Звернімо увагу на сервісний рівень та розгляньмо відмінності у роботі через різні клієнти. Лише дивлячись на діаграму класів, або головний інтерфейс можна побачити головну відмінність двох підходів – усі значення з асинхронного сервісу повертаються як CompletableFuture<>.

CompletableFuture – клас в Java для асинхронної роботи, якій містить близько 50 методів, що дозволяють створити ланцюг обробки внутрішнього компонента, який обгорнуто в цю конструкцію [21].

Для створення найпростішого фьючера можна використати методи supplyAsync та runAsync – різниця в них полягає в тому, що перший може повертати значення, а інший має тип void [21].

					IA61.260БАК.003 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

Найпростішим методом повернути результат з обгортки – є метод `get`, але при його використанні можна забути про будь-яку асинхронність, фактично – він почекає доки операція закінчиться, впевниться в тому, що вона закінчена та поверне результат – ніякого виграшу в швидкості не буде отримано.

Більш правильним способом використання фьючерів – є `callback` методи, вони починаються зі слова `then`, наприклад `thenApply`, після чого вказується яку саме дію потрібно провести над внутрішнім об'єктом в обгортці.

Великою перевагою є те, що аргументом в `callBack` можна передавати функції та бі-функції, що дозволяє створювати більш складні сценарії обробки об'єктів, а в взаємодії з `Java Stream API`, доданому в 8 версії `Java`, отримують потужний механізм для розробки асинхронних моделей.

Якщо процес виконання функції над внутрішніми об'єктами також можна робити паралельно існує механізм `then...Async`, додавання вкінці команди слова `Async`, означає, що дана операція буде виконуватись не в потоці, в якому є виклик функції, а в окремому потоці з `forkJoinPool` - це дуже зручно, коли є велика частина блокуючої, незалежної довго виконуваної логіки. Наприклад, коли отримані обчислення, необхідно записати в лог, відправити в зовнішню інтеграцію та зберегти в внутрішній БД, усі ці операції будуть мати певну затримку, яку можна мінімізувати роблячи ці дії окремими потоками.

Для того, щоб будувати складні бізнес-моделі асинхронно, в `CompletableFuture` є можливість комбінування фьючерів – це дозволяє виконувати асинхронний сценарій, коли деяка кількість інших сценаріїв будуть завершені [22]. Таким чином вибудовується перехід від неблокуючих операцій до асинхронної системи(більш детальну різницю між цими двома поняттями буде описано далі).

`CompletableFuture` надає також методи для обробки виняткових ситуацій – це методи `handle` та `exceptionally`, за своєю функціональністю `exceptionally` схожий на всім звичний `catch` – він буде виконаний, якщо очікувана

					ІА61.260БАК.003 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		

виняткова ситуація трапиться, а handle на звичний finally, його логіка буде викликана не залежно від того трапиться виняткова ситуація, чи ні.

4.5 Опис технічної реалізації тестового додатку

Основна частина реалізації являє собою два сервіси, які порівнюються між собою, блокуючий та неблокуючий відповідно, допоміжний сервіс, який генерує тестові данні для навантажувального тестування та рівня контролерів які обробляють запити та викликають сервісний рівень.

Перед тестовим проєктом поставлені дві основні цілі – це бути прикладом інтеграції та використання бібліотеки, що буде оглянуто далі, та тестувати швидкісні характеристики в реальному середовищі архітектури MVC.

Для тестування швидкодії обрані три найчастіше вживані операції, це запис в БД, отримання елемента з БД за основним ключем, та отримання елемента або елементів за додатковим індексованим полем. Також варіанти виклику цих методів будуть відрізнятися, буде проведене окреме тестування кожного методу, та послідовний виклик їх усіх в одному запиті для того, щоб продемонструвати сильні та слабкі сторони асинхронного підходу.

Клас, де використовуються асинхронні операції називається UserServiceImpl, за допомогою механізму впровадження залежностей, Spring впровадже йому DynamoDbCrudRepo бін, який є основним інтерфейсом бібліотеки, як показано на рисунку 4.17.

```
@Service
public class UserServiceImpl {

    @Autowired
    private DynamoDbCrudRepo dynamoDbCrudRepo;
```

Рисунок 4.17 – Впровадження залежності на основний інтерфейс бібліотеки

За рахунок того, що всю роботу по побудові запитів взяла на себе бібліотека, а складної бізнес логіки тестовий застосунок немає, сервіс виглядає дуже просто й є зрозумілим навіть людині далекій від програмування. Вигляд звертання продемонстровано на рисунку 4.18.

```
public CompletableFuture<PutItemResponse> saveUser(User user) {
    if (null == user || null == user.getUuid() || !user.getUuid().contains("uuid")) {
        throw new InvalidRequiredAttributeException();
    }
    return dynamoDbCrudRepo.save(user);
}
```

Рисунок 4.18 – Вигляд операції зберігання користувача

Все, що потрібно - це перевірити наявність обов'язкових полів, та передати всю роботу бібліотеці, не поглиблюючись в суть того, як саме буде опрацьовано даний запит. Такі можливості по написанню ПЗ надають великі переваги користувачам, адже клієнтський код досвідчених програмістів стає красивішим та більш коротким, а новачки не зроблять помилки при побудові запитів – все це стане роботою розробників бібліотеки. Такий підхід особливо влучний при роботі саме з хмарними застосунками, адже працюючі в хмарі, розробник не витрачає час на підтримку апаратних ресурсів, а в свою чергу бібліотеки та фреймворки дають йому змогу на рівні ПЗ скоротити час на вивчення внутрішньої будови ресурсів та інтеграцій з ними, даючи змогу розробнику зосередитись на розробці, важливій для його продукту. Так саме просто виглядають й інші запити цього сервісу, які покладають всю брудну роботу на бібліотеку.

Натомість при використанні стандартного клієнту, розробник отримує повний контроль в свої руки, що додає йому певних можливостей, які не підтримуються фреймворками та бібліотекою, однак змушує робити додаткові дії для кожного запиту. Вигляд звертання зображено на рисунку 4.19.

```
public PutItemResponse saveUserSyncClient(String tableName, Map<String, AttributeValue> fields) {
    return dynamoDbClient.putItem(PutItemRequest.builder()
        .tableName(tableName)
        .item(fields)
        .build());
}
```

Рисунок 4.19 –Конвертація об’єкту в об’єкт запиту до БД

Перша дія, яку потрібно зробити, це конвертувати транспортний об’єкт(Data Transport Object), в уніфікований об’єкт запиту до БД. В даному прикладі, це виглядає просто, однак, чим складніший запит, тим більше додаткових конфігурацій треба буде передати, та навіть для простого запиту розробнику доведеться розібратися з внутрішньою логікою роботи цього методу та усіма додатковими параметрами для того, щоб визначити потрібні вони чи ні, та які саме значення потрібно використати. Кожна додаткова логіка – це місце для помилок та додатковий код, який потрібно покрити автоматизованим або\та мануальним тестуванням, тобто збільшити ресурси на підтримку. Приклад конвертації об’єктів зображено на рисунку 4.20.

```
Map<String, AttributeValue> userAttributes = Map.of("UUID", AttributeValue.builder().s(user.getUuid()).build(),
    "USERNAME", AttributeValue.builder().s(user.getUsername()).build());

userServiceSync.saveUserSyncClient( tableName: "PROD_USER", userAttributes);
```

Рисунок 4.20 – Опис схеми для побудови запиту

Слід зазначити, що бібліотека сама описує й схему параметрів БД, базуючись на створеній метаданих, без неї цю роботу треба буде виконати в своєму коді при чому для кожного унікального запиту. В даній частині до попередньо окреслених проблем з тестуванням, підтримкою та можливими помилками додається ще одна важлива проблема – якщо контракт інтеграції змінюється на одній зі сторін, то інша сторона буде змушена вносити зміни, тобто підтримка змін в цьому запиті, та змін інтерфейсу DynamoDB лягає на плечі розробника.

Результатом використання цього підходу стає те, що навіть метод тестового застосунку, який фактично не містить бізнес-логіку з простої

					ІА61.260БАК.003 ПЗ	Арк.
						50
Змн.	Арк.	№ докум.	Підпис	Дата		

перевірки атрибутів та виклику методу інтерфейсу, перетворюється в більш складний процес для розуміння.

Після опису можливих помилок при мануальній інтеграції стає очевидною необхідність якісного тестування бібліотеки, яка виконує ці функції, не лише з точки зору швидкості, а й з точки зору коректності її роботи.

На початковому етапі тестування можна було б проводити вручну, але в довготривалій перспективі забезпечити таким чином якість досить складного коду було б дуже затратно по часу, тому тести створювалися одразу по створеному функціоналу, а частина проєкту була написана за TDD від якого згодом довелося відмовитися через нераціональність використання.

Функціональне тестування поділене на 2 основні групи – це модульні тести та інтеграційні тести. Головна різниця в них в тому, що інтеграційні тести працюють з реальною БД.

Модульні тести використовують фреймворки Mockito та JUnit. Разом вони дозволяють проводити тести одного окремого біна не піднімаючи контекст всього застосунку. Через наявність зв'язку між класами, й відповідно бінами, потрібно замінити чимось виклики до інших бінів. Саме тут стає корисним Mockito фреймворк з його ключовою анотацією @Mock, ця анотація підміняє об'єкт вказаного класу, власною реалізацією, і дозволяє гнучко перевизначати логіку роботи методів замоканого (Заміна реального об'єкту на об'єкт, яким керує тестовий фреймворк) об'єкта, й не виконуючи реальних запитів, повертати потрібну відповідь або помилку.

Приклад тесту успішного сценарію зображено на рисунку 4.21.

					ІА61.260БАК.003 ПЗ	Арк.
						51
Змн.	Арк.	№ докум.	Підпис	Дата		

```

@Test
public void parseFieldTest() throws NoSuchFieldException {
    Item ref = Item.builder().build();
    Map<String, String> fieldMapping = new HashMap<>();
    Field uuid = User.class.getDeclaredField( name: "uuid");

    KeyProperties expected = KeyProperties.builder()
        .keyType(KeyType.HASH)
        .fieldName("UUID")
        .build();

    Key key = EntityParser.parseField(ref, fieldMapping, key: null, uuid);

    assertNotNull(key);
    assertEquals( expected: 1, ref.getKeys().size());
    assertEquals(expected, ref.getKeys().get("uuid"));
}

```

Рисунок 4.21 – Тест успішного сценарію

Фактично, кожен модульний тест можна розділити на 3 основні операції – підготовка даних та очікуваних результатів тесту, виклик метода, порівняння отриманого результату з очікуваними. Для порівняння результатів використовуються методи фреймворка Junit, які надають потужний API для перевірки виконання різноманітних умов.

Гарною практикою тестування ПЗ є створення тестів не лише на один успішний очікуваний сценарій, а й покриття сценаріїв, що викликають очікувані помилки.

```

@Test(expected = RuntimeException.class)
public void parseFieldWithMultipleKeyAnnotationOnOneEntity() {
    Key [] keys = mock(Key[].class);
    keys [0] = mock(Key.class);
    keys [1] = mock(Key.class);
    Field fieldWith2Keys = mock(Field.class);
    doReturn(keys).when(fieldWith2Keys).getAnnotationsByType(Key.class);
    EntityParser.parseKey( item: null, key: null, fieldWith2Keys);
}

```

Рисунок 4.22 – Тест сценарію з виключною ситуацією

Змн.	Арк.	№ докум.	Підпис	Дата

Тест на рисунку 4.22 очікує виняткову ситуацію як результат виклику, це перевірка правильності логіки при неуспішних сценаріях, коли користувач неправильно створить анотації над своїми класами, та 1 поле буде одночасно мати 2 анотації Key, що є помилкою. Тест перевіряє правильну реакцію бібліотеки на помилковий сценарій.

Основний клас бібліотеки DynamoDbCrudRepoImpl має сто відсоткове покриття тестами, що включають модульні тести на рівні бібліотеки, та інтеграційні на рівні тестового додатку.

```
@RunWith(MockitoJUnitRunner.class)
public class DynamoDbCrudRepoImplTest {

    @Mock
    private DynamoDbAsyncClient dynamoDbAsyncClient;
```

Рисунок 4.23 – Заміна клієнту БД

Для того щоб імітувати відповідь з БД використовується вище згадана анотація @Mock (зображена на рисунку 4.23) за допомогою якої повертається значення необхідне для проведення певного сценарію тестування.

Інтеграційні тести, які працюють з реальною БД, на відміну від модульних піднімають ApplicationContext, як і реальний застосунок, але можливо вказати, які біни увійдуть до цього контексту – тобто він може відрізнитися від оригінального. Зазвичай у контекст входять декілька класів з тестовими налаштуваннями, які відповідають за якийсь логічно завершений сценарій застосунку. Налаштування контексту показано на рисунку 4.24.

```
@RunWith(SpringRunner.class)
@SpringBootTest
public class UserServiceImplTest {

    @Autowired
    private UserServiceImpl userService;
```

Рисунок 4.24 – Підняття ApplicationContext в тесті та використання реальних бінів

Щоб імітувати відповіді, які реально виконати неможливо, або зайво в певному сценарії, замість анотації @Mock використовується анотація @MockBean, головна відмінність якої полягає в тому, що об'єкт, який анотовано цією анотацією витісняє з контексту всі інші біни такого ж типу та сам додається до контексту.

4.6 Реалізація навантажувального тестування

Навантажувальне тестування реалізовувалося з двох сторін. Зі сторони тестового застосунку були створені контролери для прийняття HTTP запитів, а для імітації одночасного використання сервісу була написана додаткова програма на Scala з використанням фреймворку Gatling.

Лістинг Scala застосунку – ДОДАТОК В.

Мова Scala була обрана з метою використання орієнтованого під цю мову фреймворка - Gatling.

Навантажувальне тестування складається з 4 функціональних сценаріїв, що для синхронного та асинхронного варіанту відповідно дають 8 сценаріїв.

В тестуванні використовуються наступні сценарії:

- отримання об'єкту User по id синхронно;
- отримання об'єкту User по id асинхронно;
- запис нового об'єкту User синхронно;
- запис нового об'єкту User асинхронно;
- стресовий сценарій з перевірки наявності об'єкту User по id, запису нового об'єкту User, якщо такого немає, отримання цього User по індексованому полю – усі операції синхронні;
- стресовий сценарій з перевірки наявності об'єкту User по id, запису нового об'єкту User, якщо такого немає, отримання цього User по індексованому полю – усі операції асинхронні;
- сценарій зі створення об'єкту User та виконання двох ресурсномістких операцій асинхронно;

					ІА61.260БАК.003 ПЗ	Арк.
						54
Змн.	Арк.	№ докум.	Підпис	Дата		

– сценарій зі створення об'єкту User та виконання двох ресурсномістких операцій синхронно.

Такі сценарії обрані через те, що це найчастіше використовувані операції відносно БД, та тому, що на їх прикладі можна якомога краще продемонструвати слабкі та сильні сторони асинхронних підходів.

З боку тестового додатку ця логіка реалізована в UserController.class. Таблиця REST контролерів для інтеграції тестуемого та тестового додатку наведена нижче.

Таблиця 4.1 – API застосунку

Метод запит у HTTP	URL адреса	Параметри	Призначення
1	2	3	4
GET	/user/{user_id}	user_id: унікальний ідентифікатор користувача	Отримати User по ID асинхронно
GET	/user/sync/{user_id}	user_id: унікальний ідентифікатор користувача	Отримати User по ID синхронно
POST	/user/create	username: ім'я юзера uuid: id юзера	Створити нового User асинхронно
POST	/user/sync/create	username: ім'я юзера uuid: унікальний ідентифікатор користувача	Створити нового User синхронно

Таблиця 4.1(закінчення)

POST	/user/fullsycle	username: ім'я юзера uuid: унікальний ідентифікатор користувача	Перевірити, що User не існує по id, створити нового, повернути його після цього по username асинхронно
POST	/user/sync/fullsycle	username: ім'я юзера uuid: унікальний ідентифікатор користувача	Виконати операції /user/fullsycle синхронно
POST	/user/create/operations	username: ім'я юзера uuid: унікальний ідентифікатор користувача	Створити нового User та виконати додаткові важкі операції - асинхронно
POST	/user/sync/create/operations	username: ім'я юзера uuid: унікальний ідентифікатор користувача	Створити нового User та виконати додаткові важкі операції - синхронно

Виконання кожного зі сценаріїв можна поділити на 3 кроки – це створення емуляції, емуляція та візуальне зображення результатів емуляції.

В процес створення емуляції входить процес створення віртуальних користувачів у вказаній кількості з заданими параметрами, які будуть робити запити. Дуже важливо, що ця дія робиться окремо, а не під час тестування системи – при такому підході процес створення об'єктів необхідних для тестування не впливає на спільні ресурси з системою яка тестується, що при локальному тестуванні відіграє дуже значну роль.

Процес створення тестових даних в процесі тестової емуляції – це дуже важливий процес. Зрозуміло, що проводити тестування на “hardcoded” даних не має ніякого сенсу, бо алгоритмічна складність діставання такого елемента буде $O(1)$, тому ця операція буде швидкою з будь-яким алгоритмом обробки. З іншого боку, для пошуку багів – характеристики кожного тестового об’єкту повинні бути відомі. Також важливу роль відіграє рівномірність або нерівномірність розподілу даних для ключів, адже як було вказано раніше, для кожного ключа вираховується хеш-код по якому виконується пошук.

Щоб забезпечити виконання всіх вимог, необхідно було створити окремий сервіс в тестовому застосунку `GenerateUsersForLoadTestService.class` для того, щоб емітувати розподіл даних близько до реальних систем.

Поля по яким вираховують хеш мають статичну частину та випадково згенеровану частину з рівномірним розподілом, як показано на рисунку 4.25.

```
String uuid = "uuid-"+i+"_"+random.nextLong();
```

Рисунок 4.25 – Приклад генерації поля, по якому буде вираховуватись хеш для пошуку в БД

Для тестування було створено набір даних з 1 мільйона тестових об’єктів (ДОДАТОК Г), які використовуються циклічно (тобто якщо під час тестів буде використано об’єкт номер 999 999, наступним буде елемент – 0, а не помилка). Ці об’єкти були поміщені в тестовий .csv файл, й додані до застосунку, що виконує тестування, у вигляді “feeder” – це визначення ресурсу в Gatling фреймворку, звідки будуть братися тестові данні, це може бути кеш, БД, зовнішня система або файл як в нашому випадку. Мільйон рівномірно розподілених тестових об’єктів забезпечують здорове тестування з очікуваною логарифмічною складністю пошуку об’єкта.

Сам процес тестування навантаження є простим і фактично реалізований фреймворком повністю, все що потрібно зробити – налаштувати. В

налаштуваннях вказується кількість одночасних віртуальних користувачів, час проведення тестування, мінімальний та максимальний очікуваний час виконання одного запиту, очікуваний параметр та відсоток успіху. Налаштування зображені на рисунку 4.26.

```
object PerfTestConfig {
    val baseUrl = getStringOrElse("baseUrl", "http://localhost:8080")
    val requestPerSecond = getAsDoubleOrElse("requestPerSecond", 100f)
    val durationMin = getAsDoubleOrElse("durationMin", 1.0)
    val meanResponseTimeMs = getIntOrElse("meanResponseTimeMs", 10)
    val maxResponseTimeMs = getIntOrElse("maxResponseTimeMs", 500)
}
```

Рисунок 4.26 – Налаштування системи, що виконує тестування

В сценарії тестування отримання об'єкту User по id – суть полягала у проведенні лише однієї операції, отримання з БД об'єкту User з id з запиту з використанням синхронного клієнту та розробленої бібліотеки, що базується на асинхронному клієнті.

З заздалегідь підготовленого файлу було створено користувачів, за допомогою яких (вибираючи id випадковим чином) створювались запити.

Як можна побачити на графіку зображеному на рисунку 4.27, побудованому Gatling фреймворком, більше 60% запитів були виконані в часовому проміжку 0-19 мс, враховуючи те, що час витрачався не тільки на запит до БД, а й на інші процеси (створення об'єктів, обробка HTTP запиту, тощо) можна сказати, що 60% запитів вклалися в обіцяну AWS швидкість 0-9 мс при користуванні синхронним клієнтом.

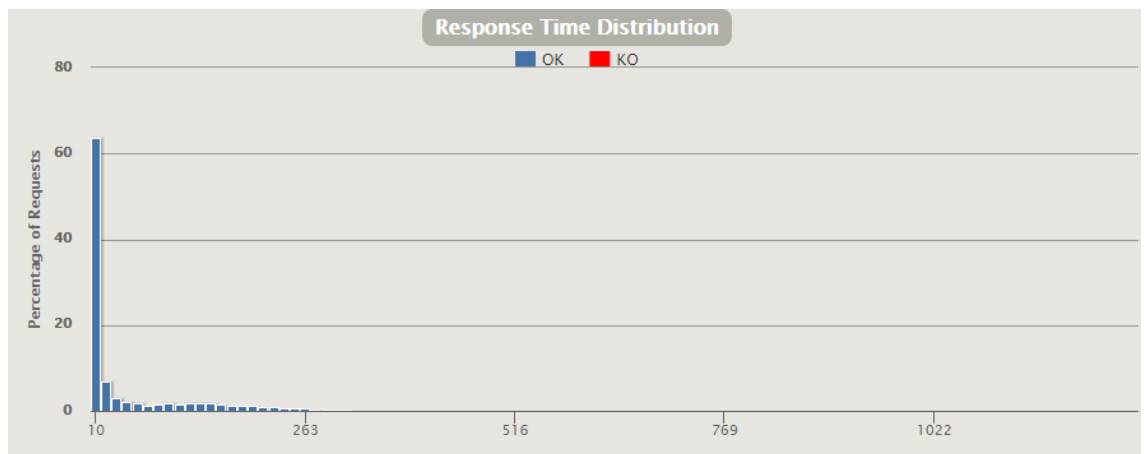


Рисунок 4.27 – Розподіл відповідей за часом для синхронного клієнту

Як можна побачити на рисунку 4.28, фреймворк створює піки навантажень, тобто при виставленій кількості 300 одночасних запитів – фреймворк буде створювати піки в 500-1000 одночасних запитів, для того, щоб показати як система реагує на піки й відповідно наскільки вона може бути масштабована. Як видно на розподілі, у синхронного клієнта очікувана, близька до лінійної, залежність від кількості запитів при виконанні лише однієї операції. Виходячи з цього можна сказати, що синхронний клієнт AWS розроблено якісно й системи, які його використовують можуть якісно масштабуватися горизонтально.

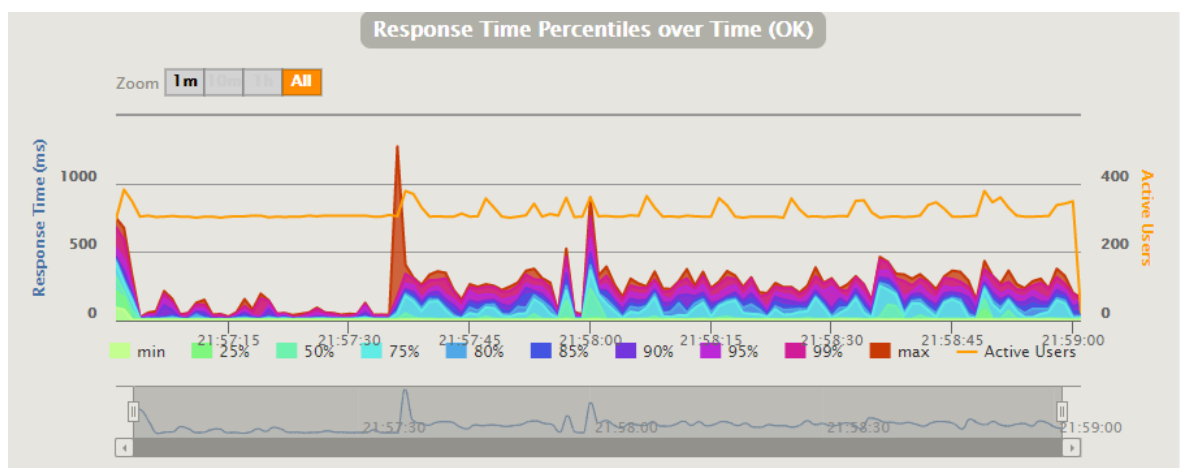


Рисунок 4.28 – Розподіл пікових навантажень для синхронного клієнту

Змн.	Арк.	№ докум.	Підпис	Дата

Як видно на рисунку 4.29, 55% запитів вклалися в очікуваний результат 0-19 мс, це менше у порівнянні з використанням клієнту. Очевидно, що додатковий час був витрачений на автоматизовану побудову запиту, потрібно було відшукати потрібну мета-дату, на базі метадати конвертувати отриманий об'єкт в запит до БД.

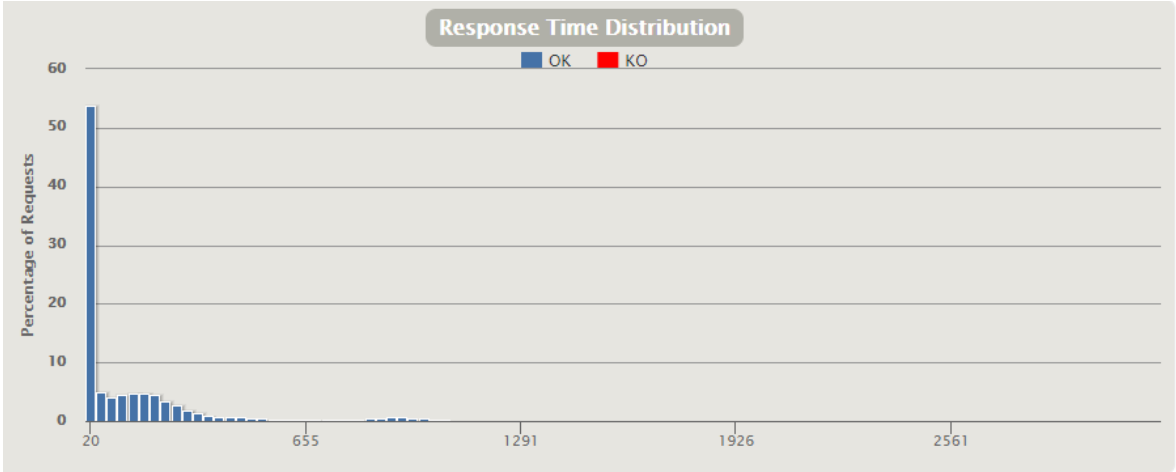


Рисунок 4.29 – розподіл відповідей за часом з використанням бібліотеки

Буде не до кінця вірно сказати, що бібліотека впоралася гірше, адже вона виконала додаткову роботу по автоматичній генерації запиту, а асинхронність фактично була зведена на нівець(бо в цьому сценарії виконувалась лише одна операція). Висновком може бути те, що розроблена бібліотека не гірше синхронного клієнту, а програш в оптимізації у 5% - це плата за спрощену розробку та підтримку.

По графіку розподілу піків навантаження, зображеному на рисунку 4.30, можна побачити, що асинхронна система краще на них реагує – навантаження розподіляється за нормальним законом розподілу і швидкодія обмежується лише ресурсними апаратними можливостями. На графіку видно, що час запитів розподілився більш рівномірно – тобто всі запити стали в середньому довгими, а не один запит швидкий, а інший – дуже довгий, як при синхронному клієнті.

Таке покращення досягається шляхом виконання запитів до БД - кожного в окремому потоці, й не блокування потоків, що відповідають за обробку HTTP запитів.

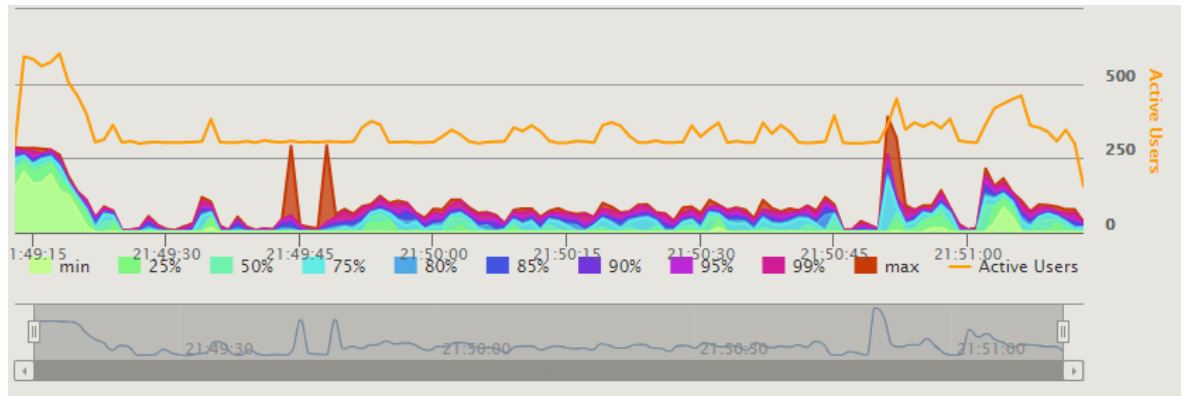


Рисунок 4.30 – Розподіл пікових навантажень для асинхронного клієнту

Отримані результати по цим сценаріям тестування підтверджують очікування та виконання поставлених цілей на початку роботи - на одиноких запитах, що не дають проявляти сильні сторони асинхронності, бібліотека працює не гірше за синхронний клієнт.

При тестуванні запису об'єкта User в БД - основною метою є отримання схожих показників до показників отримання об'єкта User по id, в сценарій входить єдина операція по запису юзера до БД.

Варто зазначити, що в обох випадках в незалежності від клієнта, запис буде не до кінця оптимальним, бо обидва тестові поля тестового об'єкту проіндексовані, тобто БД витратить додатковий час на створення індексів.

На рисунку 4.31 видно, що розподіл часу обробки операцій створення дуже близький до зчитування, більше 60% операцій вклалися в показник. Близькість операцій по часу виконання свідчить про коректну реалізацію системи та здорове тестування, адже за документацією AWS, DynamoDB – однаково швидка для обох операцій.

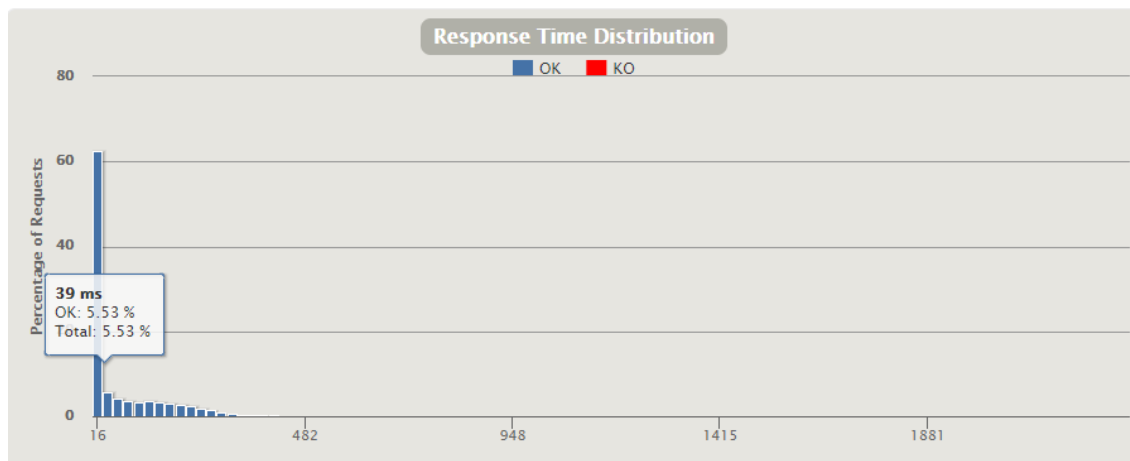


Рисунок 4.31 – Розподіл відповідей за часом з використанням клієнту

На рисунку 4.32 видно, що реакція синхронного клієнту на пікову кількість запитів створення гірша за лінійну й відповідно гірша за реакцію на піки зчитування – головною причиною є саме додатковий час, що витрачається на індексацію. Важливо розуміти, що зі збільшенням системи, тобто зі збільшенням розміру полів та їх вмісту та загальної кількості запитів – ситуація буде погіршуватися.

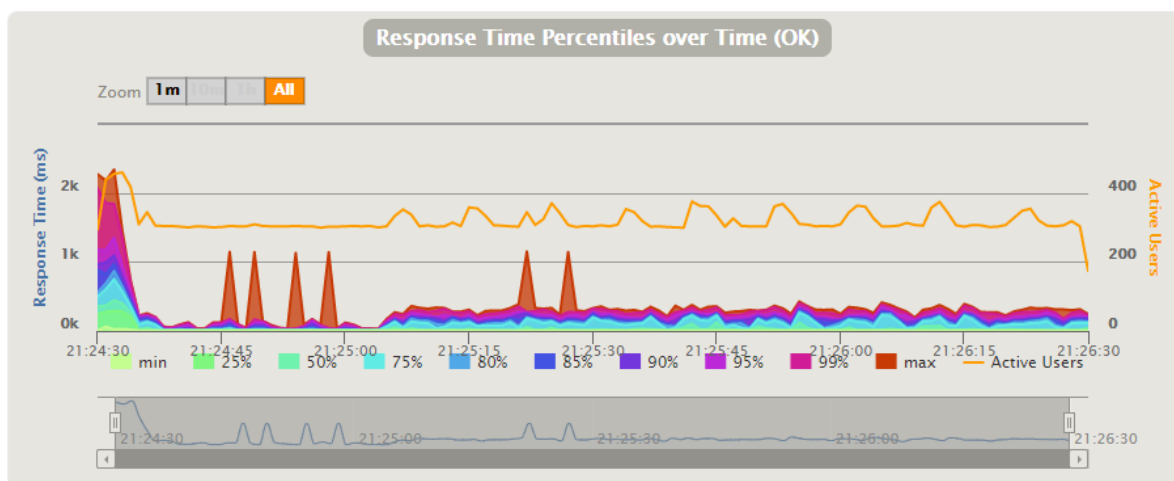


Рисунок 4.32 – Розподіл пікових навантажень для клієнту

Для DynamoDB взагалі властиві проблеми з обробкою піків записів, адже ніхто не хоче підвищувати WriteCapacityUnits(ємність запису), бо це збільшує вартість таблиці, заради поодиноких стрибків навантаження, однак це є фактично єдиним рішенням для масштабування системи на запис.

Змн.	Арк.	№ докум.	Підпис	Дата

Дивлячись на рисунок 4.33, можна побачити, що для операції запису асинхронний клієнт показав себе краще, порівняно з читанням відносно синхронного клієнту, час що бібліотека витратила на додаткові операції, був компенсований економією часу, коли система не очікувала завершення виконання більш тривалої операції на стороні БД.

Розподіли часу відповіді для синхронного клієнту та бібліотеки майже еквівалентні на всіх проміжках, що є гарним результатом, який навіть перевершує очікування від бібліотеки.

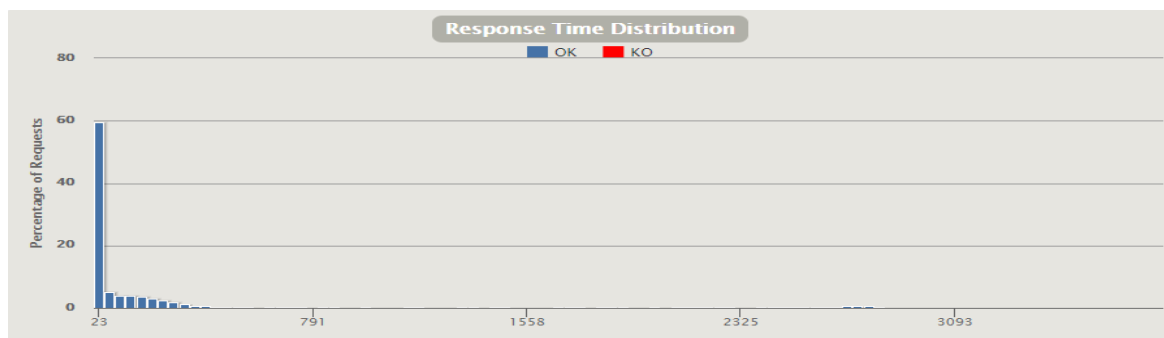


Рисунок 4.33 - Розподіл відповідей за часом з використанням бібліотеки

Розподіл пікових навантажень асинхронного клієнту зображено на рисунку 4.34, як і для операції зчитування, значно кращий у реалізованій бібліотеки порівняно з синхронним клієнтом – на графіку майже відсутні червоні, оранжеві та фіолетові зони, що відображають значну затримку, а навантаження було рівномірно розподілене між всіма запитами (для синхронного клієнту середнє значення зеленого кольору – 16 мс, а для асинхронного – 23 мс).

Змн.	Арк.	№ докум.	Підпис	Дата

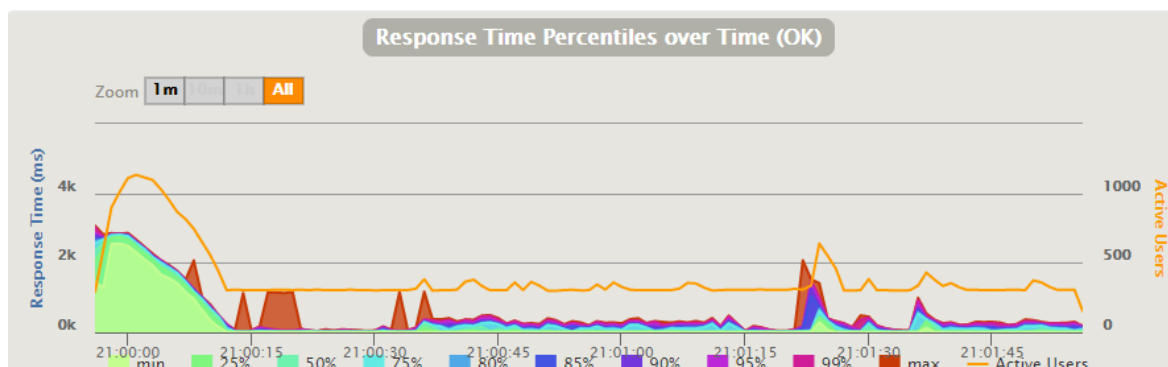


Рисунок 4.34 – Розподіл пікових навантажень для асинхронного клієнту

По результатам тестування цього сценарію бібліотека показала себе еквівалентною по швидкодії синхронній системі при очікуваному навантаженні та значно кращою при пікових навантаженнях і масштабуванні.

Ресурсномісткий сценарій створений для того, щоб показати очікуване використання асинхронної системи та її найсильніші сторони. Алгоритм сценарію полягає в тому, щоб подавати на запис випадкових тестових юзерів, записувати їх в БД, додавати в колекцію `ConcurrentHashMap`, а якщо такий елемент вже існував замінити його там, повторно обчислюючи хеш-код старого об'єкта і повертаючи його. Усі ці складнощі були додані навмисно, щоб імітувати реальні затрати часу на виконання складної бізнес-логіки.

Важливим показником в цьому сценарії, стало саме те, що синхронний клієнт ніяким чином не виконав 1397 запитів з 36 000 (розподіл зображено на рисунку 4.35), це означає, що за налаштований максимальний час (1200 мс) віртуальний користувач не зміг отримати ніякої відповіді від системи взагалі. Це дуже показово, що зі збільшенням складності системи, у користувачів синхронного клієнта не залишається механізмів покращення швидкодії – лише горизонтальне масштабування й відповідно більші економічні витрати, або збільшення часу очікування для користувачів, тобто зменшення якості системи.

Змн.	Арк.	№ докум.	Підпис	Дата

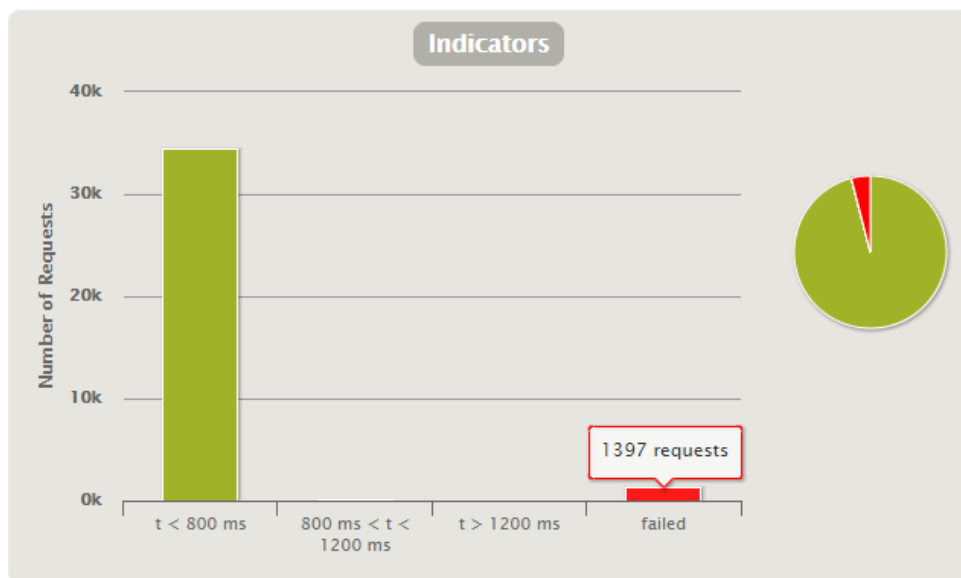


Рисунок 4.35 – Розподіл виконаних та невиконаних запитів з використанням синхронного клієнту

Варто відзначити, що все ж ті запити, які були оброблені – були оброблені досить швидко – більша частина (майже 80% запитів) були оброблені в діапазоні 0-29 мс, як видно на рисунку 4.36.

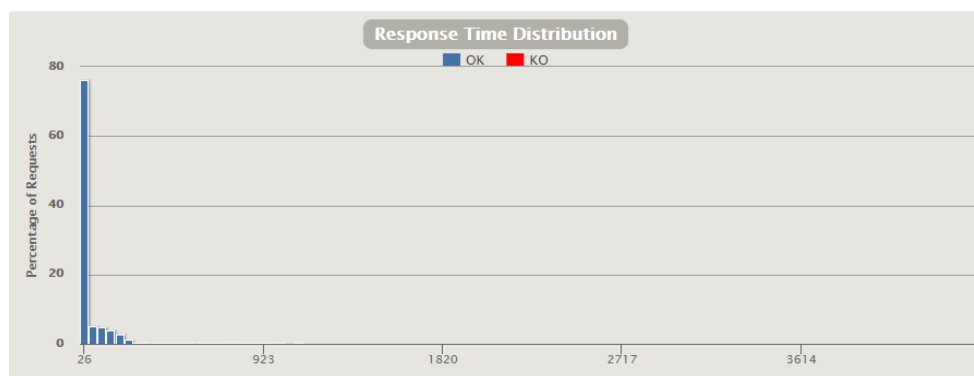


Рисунок 4.36 - Розподіл відповідей за часом з використанням синхронного клієнту

Розподіл виконаних пікових навантажень для синхронного клієнту зображено на рисунку 4.37. Для цього сценарію він виглядає краще ніж для інших синхронних сценаріїв, але це лише через те, що 1397 запитів не були виконані зовсім.

Змн.	Арк.	№ докум.	Підпис	Дата

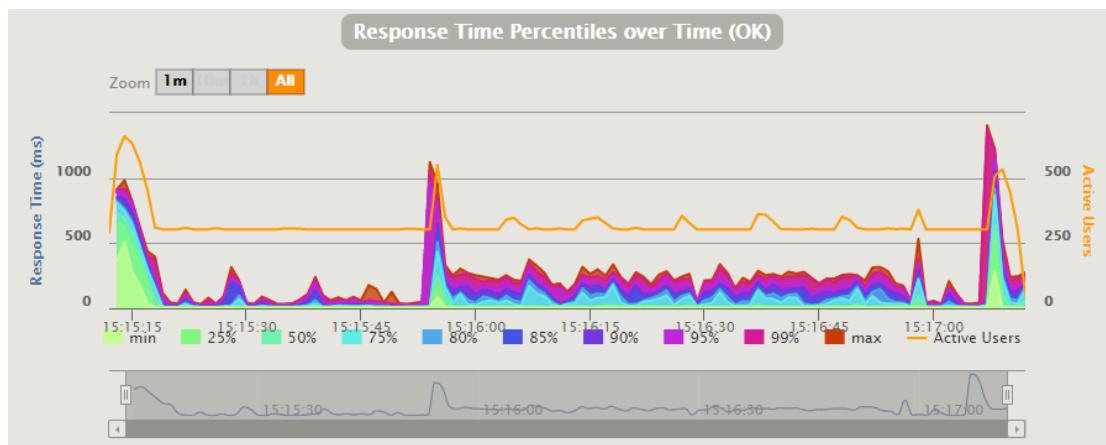


Рисунок 4.37 – Розподіл пікових навантажень для клієнту

Для асинхронного клієнту кількість невиконаних запитів є на третину меншою (розподіл продемонстровано на рисунку 4.38) й, що саме головне, природа цих помилок інша (буде описано в наступному тестовому сценарії) й вони такі, що їх можна вирішити шляхом оптимізації, а не масштабування чи збільшення часу очікування.

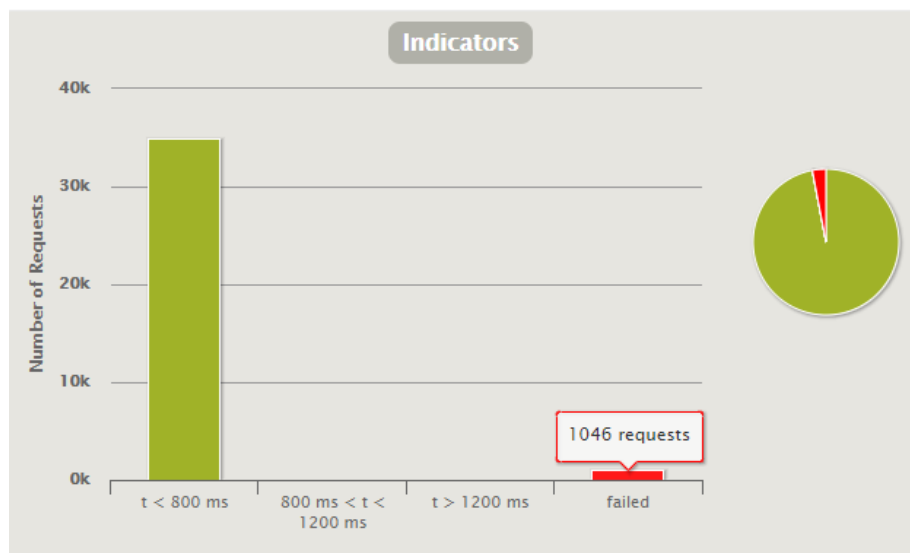


Рисунок 4.38 –Розподіл виконаних та невиконаних запитів з використанням асинхронного клієнту

Запити, які були виконані, виконувалися в середньому з тією ж швидкістю, що і при синхронному клієнті, 70% запитів лежать в діапазоні 0-

Змн.	Арк.	№ докум.	Підпис	Дата

20мс, а 6% запитів лежать в діапазоні 20-55мс. Розподіл зображено на рисунку 4.39.

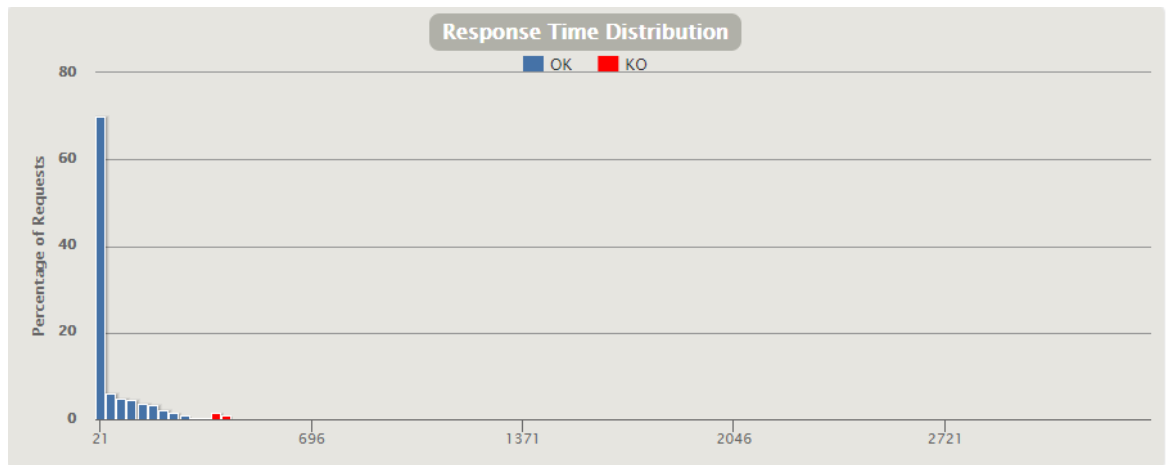


Рисунок 4.39 – Розподіл відповідей за часом з використанням бібліотеки

Дуже важливим показником – є наявність на графіку червоних стовпчиків – це час відповіді системи при помилці. Це дуже важливо з точки зору того, що навіть коли система з використанням бібліотеки не змогла повністю виконати сценарій, вона все ж обробила цю помилку і змогла відповісти, що свідчить про іншу природу помилки і більш високу якість роботи системи порівняно з роботою на синхронному клієнті.

Як і в інших випадках, система заснована на асинхронних викликах розподіляла пікове навантаження між всіма потоками, що дозволило більш ефективно протистояти піковому збільшенню навантаження. Розподіл зображено на рисунку 4.40.

Результатом тестування цього сценарію є те, що розроблена бібліотека продемонструвала свої сильні сторони – а саме швидкодію та високу надійність при роботі з вищим за очікуваний навантаженням.

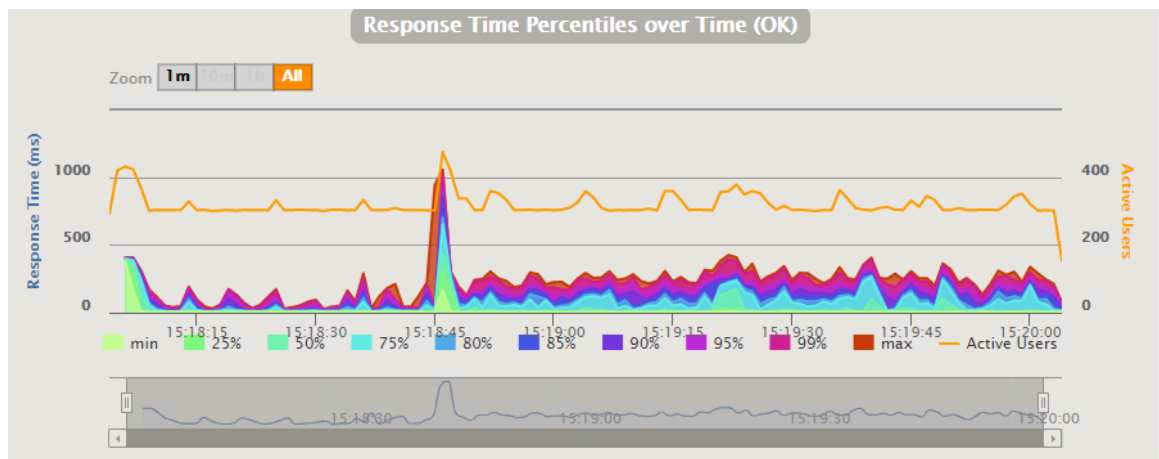


Рисунок 4.40 – Розподіл пікових навантажень для асинхронного клієнту

Стресові сценарії були створені для того, щоб продемонструвати поведінку двох систем при стресовому навантаженні, яке система не здатна витримати й показати складнощі масштабування асинхронної системи.

В цьому сценарії на кожний HTTP запит віртуального користувача система буде виконувати 3 запити до БД – 2 зчитування та 1 запис, тобто фактично навантаження на систему та БД потроєне й націлене саме на взаємодію застосунку та БД. Окрім іншого це тестування допоможе відкрити природу помилок минулого тестового сценарію для асинхронного клієнту.

Синхронний клієнт в стресовому режимі зміг виконати 8616 запитів з 18000, як показано на рисунку 4.41, (час тестування довелося зменшити вдвічі, бо при тривалому тестуванні ОС тестового ноутбуку як і застосунок переставали відповідати).

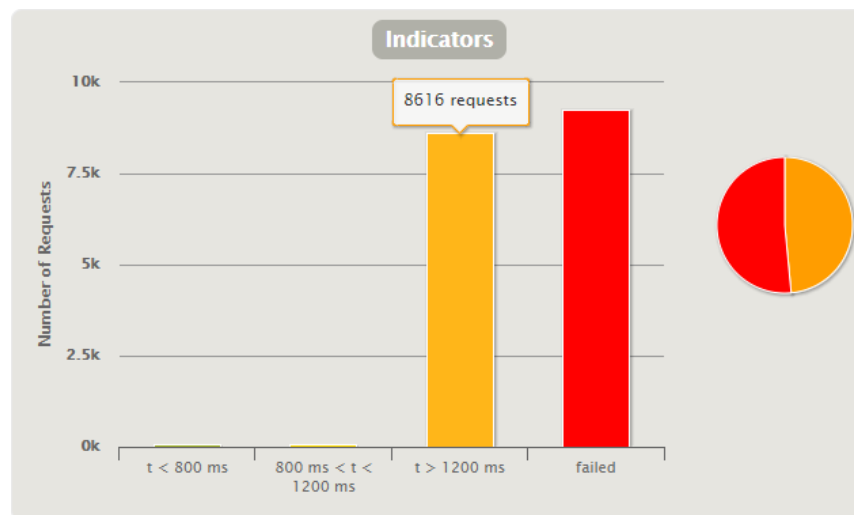


Рисунок 4.41 – Розподіл виконаних та невиконаних запитів з використанням синхронного клієнту

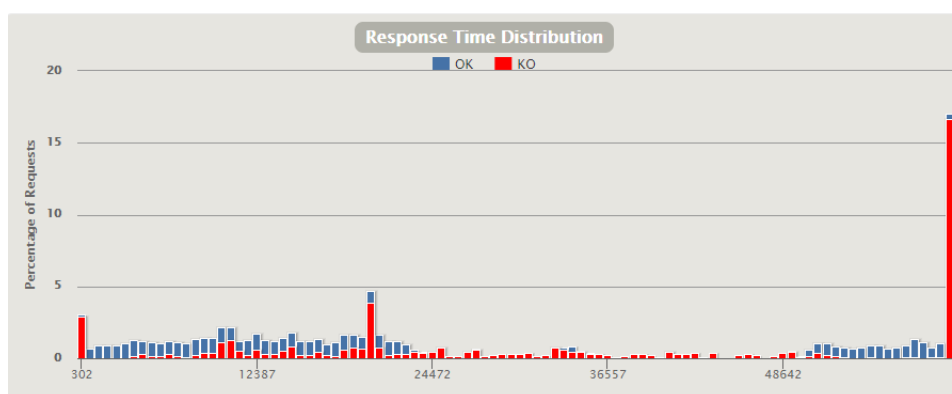


Рисунок 4.42 – Розподіл відповідей за часом з використанням клієнту

Менше 1% запитів змогли задовольнити умовам системи при роботі зі стресовим навантаженням, як видно на рисунку 4.42.

Як видно на рисунку 4.43 на певному етапі в системі назбиралося близько 5 тисяч віртуальних користувачів на секунду(відповідно 15 тисяч запитів до БД за секунду), що завмерли в очікуванні, бо не отримували відповіді, й система перестала відповідати після цього зовсім. 2 зелені стовпці після цього це відповідь по HTTP протоколу, що час відповіді від сервера добіг до кінця.

Змн.	Арк.	№ докум.	Підпис	Дата

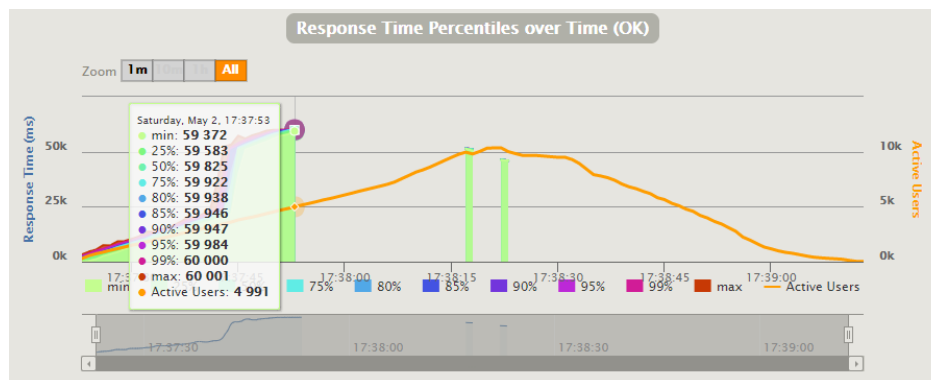


Рисунок 4.43 – Розподіл пікових навантажень для клієнту

На рисунку 4.44 видно, що система з використанням бібліотеки в стресовому режимі змогла виконати успішно 4218 запитів, що вдвійчі менше за показники синхронного клієнту.

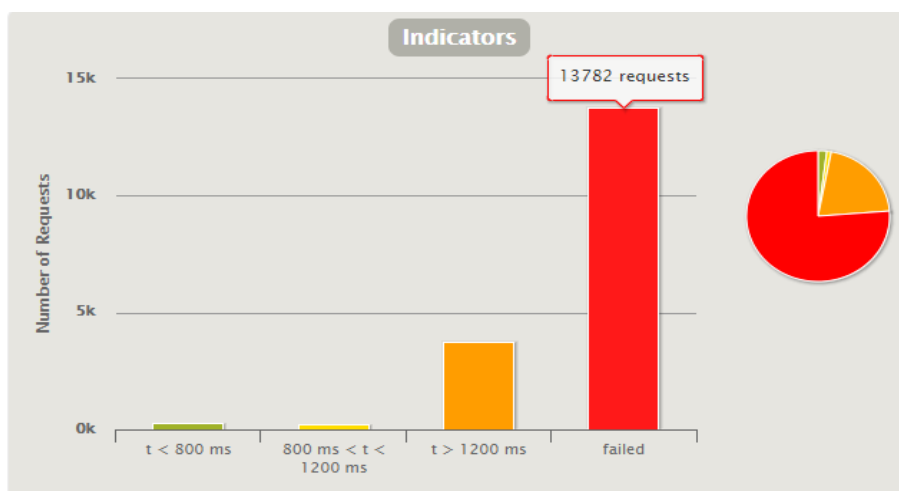


Рисунок 4.44 – Розподіл виконаних та невиконаних запитів з використанням асинхронного клієнту

По-перше, це пов'язано з алгоритмом рівномірного розподілу затримки між всіма потоками, що призвело до того, що значна частина потоків виконалася, але із затримкою – тому вони вважаються також неуспішними.

По-друге, це пов'язано з необхідністю ручного керування життєвим циклом асинхронних потоків, на відміну від єдиного потоку в синхронному режимі, яким керує Spring фреймворк. Керування потоком необхідне через те, що головний потік, який породжується при обробці HTTP запиту буде

Змн.	Арк.	№ докум.	Підпис	Дата

звільнено Spring фреймворком, коли добіжить кінця час максимального очікування HTTP запиту, однак саме цей потік керує потоками, що були створені для асинхронних запитів, тому, якщо Spring «вб'є» цей потік раніше, ніж цей потік «вб'є» дочірні потоки, то в системі буде витік пам'яті. Для запобігання витіку пам'яті механізм фьючерів надає механізм вбивання потоків через максимальний час очікування, але за це доводиться платити асинхронністю (кожна операція по перевірці чи потік завершив свою роботу є блокуючою). Фактично, алгоритм реалізовано таким чином, щоб «вбивати» дочірні потоки за декілька мілі секунд до того, як Spring «вб'є» батьківський потік.

Якщо система починає працювати в стресовому режимі, коли кожна операція завершується з часом близьким до максимального, то час витрачений на перевірки для запобігання витоку пам'яті стає більшим ніж вииграш від асинхронності й система працює повільно й ненадійно (рисунок 4.45). Саме тому при масштабуванні асинхронної системи важливо проєктувати так, щоб пікове навантаження не переростало в постійне стресове навантаження без додаткової оптимізації чи горизонтального масштабування.

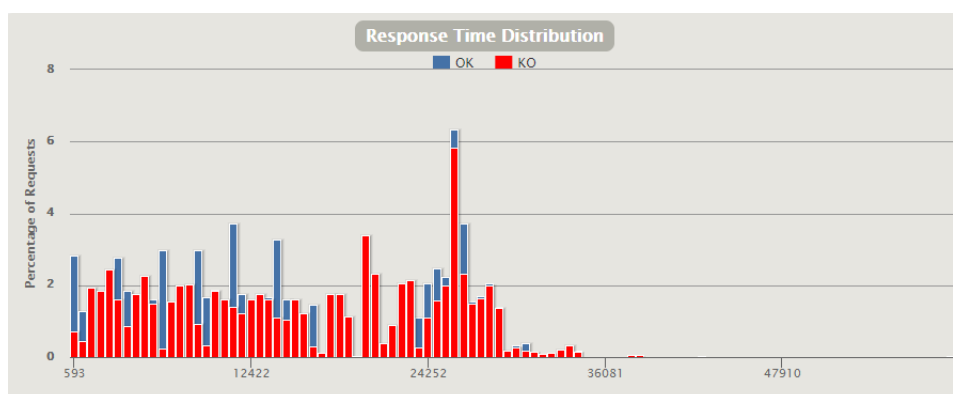


Рисунок 4.45 – Розподіл відповідей за часом з використанням бібліотеки

Змн.	Арк.	№ докум.	Підпис	Дата

При використанні бібліотеки вдалося забезпечити виконання 3% запитів успішно з заданою якістю (до 1200 мс), як видно з графіку зображеного на рисунку 4.45.

При використанні бібліотеки система не “померла” під створеним навантаженням, а продовжувала працювати, хоч і не оптимально, обробляючи ті запити, які встигає, і вбиваючи ті, які не встигають виконатися (рисунок 4.46).

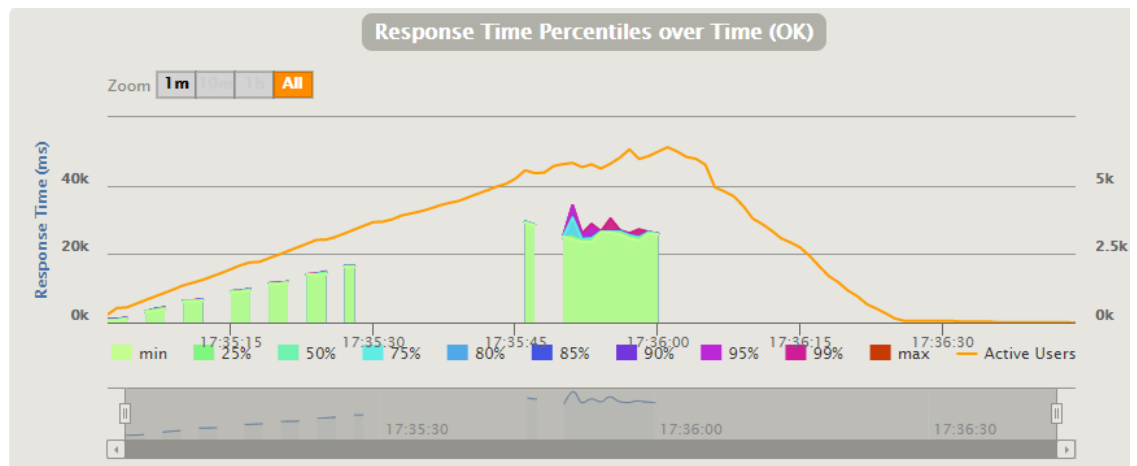


Рисунок 4.46 – Розподіл пікових навантажень для асинхронного клієнту

На рисунку 4.47 видно, що система навіть змогла пережити пік й продовжити відповідати не тільки про помилку, а й про успішні операції.

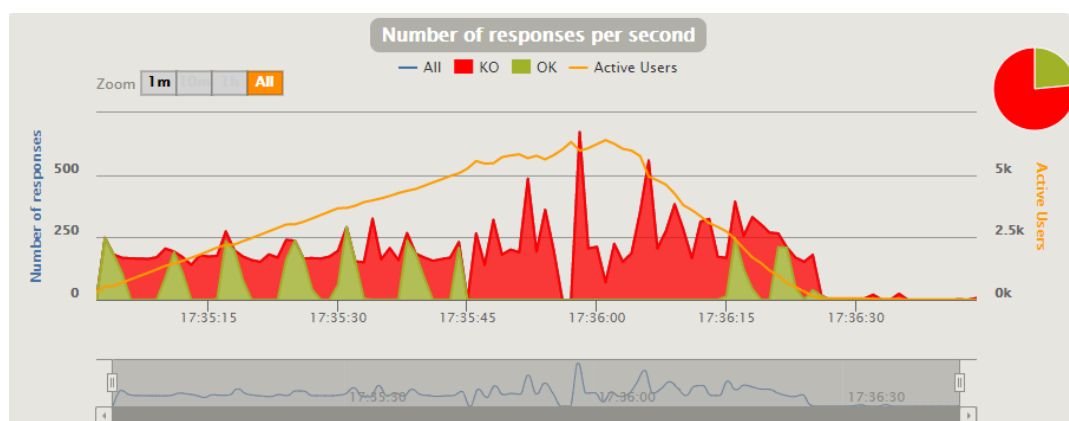


Рисунок 4.47 – Кількість відповідей в секунду

Змн.	Арк.	№ докум.	Підпис	Дата

4.7 Види блокування

Як було видно при проведенні навантажувального тестування на візуалізованих графіках, процес блокування – відбувався в обох випадках, однак виглядав по-різному. Головна причина цьому – це наявність двох видів блокування таких, як блокування CPU та I/O блокування.

I/O блокування - вхід / вихід (ІО) стосується взаємодії з такими пристроями, як жорсткий диск, мережа або база даних. Зазвичай все, що не відбувається в процесорі, називається ІО. Коли ви викликаєте API, який запитує дані з ІО, ви не отримаєте відповідь миттєво, але з деякою затримкою. Ця затримка може бути дуже невеликою для запиту файлу на жорсткому диску і набагато довше, коли запит даних із мережі. Це пов'язано з тим, що дані, які ви запитуєте з пристроїв вводу-виводу, повинні довше доставлятися абоненту [23]. Тут потік блокується, оскільки він повинен чекати повернення даних з джерела вводу-виводу, наприклад, мережі або жорсткого диска. Ядро помітить, що немає даних з ІО, і тому переведе потік у стан "сну". Отже, при блокуванні ІО-блокування потік не виконується активно на процесорі.

CPU блокування - у цьому випадку потік блокується через якесь інтенсивне завдання процесора, виконання якого, займає більше часу, ніж «миттєво». Наприклад, при генерації простих чисел або візуалізації 3d-моделі. При блокуванні процесора потік блокується, оскільки він активно виконується на процесорі [23].

Отже при використанні асинхронної бібліотеки під час піків було продемонстроване CPU блокування через використання складних операцій, час виконання яких на процесорі є не атомарним. При використанні синхронного клієнту блокування, й відповідно піки, були пов'язані саме з необхідністю очікування повернення результату з БД (I/O).

CPU блокування є нормальним процесом і може ефективно вирішуватись правильними налаштуваннями ресурсів контейнеру та Java застосунку й методом горизонтального масштабування, якщо ресурс CPU

					ІА61.260БАК.003 ПЗ	Арк.
						73
Змн.	Арк.	№ докум.	Підпис	Дата		

вичерпано повністю, в той час як I/O блокування призводить до нераціонального використання наявних ресурсів й має вирішуватись саме переходом на неблокуючі технології.

Розроблений проєкт дасть змогу користувачам DynamoDB вирішити проблему I/O блокування.

Висновки до розділу 4

Розроблений додаток, який інтегрувався з бібліотекою й виконував роль прикладу по інтеграції й тестового застосунку.

Обране середовище розгортання тестової БД, шляхом обчислення економічних ресурсів та принципу раціональності.

Створена тестова БД.

Проведене функціональне тестування розробленого додатку, що включало модульні, інтеграційні та мануальні тестові заходи.

Оглянуті теоретичні аспекти асинхронності та неблокуючих операцій, а також шляхи реалізації рішень, що використовують ці підходи, в Java екосистемі.

Розроблено 4 сценарії, 2 з яких є не оптимальними для асинхронного підходу, 1 оптимальна, та 1 стресова.

Проведене end-to-end сценарій навантажувального тестування в еквівалентній ситуації до реального користувача.

З метою забезпечення високої якості тестування була написана окрема програма на Gatling, яка імітує одночасні виклики API тестового застосунку.

З боку тестового застосунку були реалізовані сценарії з використанням синхронних та асинхронних операцій.

По результатам тестування були порівняні наступні показники:

- надійність;
- швидкодія (час відгуку);
- обробка помилок;

					ІА61.260БАК.003 ПЗ	Арк.
						74
Змн.	Арк.	№ докум.	Підпис	Дата		

– масштабованість.

За результатами тестування було доведено:

- неефективність використання асинхронного підходу для однієї операції та для дуже простих операцій;
- вищу ефективність та надійність асинхронності при роботі з декількома важкими операціями;
- необхідність використання правильних підходів при масштабуванні асинхронних додатків.

Були розглянуті проблеми:

- блокування по CPU;
- блокуючий I/O;
- витік пам'яті.

Запропоноване рішення для розглянутих проблем.

					ІА61.260БАК.003 ПЗ	Арк.
						75
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

Була розроблена бібліотека для генерації запитів до бази даних заснованих на асинхронних операціях.

Розроблений проєкт представляє рішення, що задовольняє вимогам ринку з використанням сучасних технологій та орієнтацією на хмарні рішення через зростання їх популярності. Проєкт уникнув недоліків існуючих рішень, які були виявлені під час аналізу.

Розроблений зручний інтерфейс взаємодії з бібліотекою для користувачів, який дає можливість розробникам одразу використовувати бібліотеку, а не вивчати її внутрішню будову.

Була закладена можливість додавання нових інтерфейсів взаємодії для подальшого розширення функціоналу для нереалізованих на даному етапі сценаріїв, що будуть додані в наступних версіях.

Розроблено додаток, який інтегрувався з бібліотекою й виконував роль прикладу по інтеграції й тестового застосунку.

Було обране середовище розгортання тестової БД, шляхом обчислення економічних ресурсів та принципу раціональності.

Проведене функціональне тестування розробленого додатку, що включало модульні, інтеграційні та мануальні тестові заходи.

Були оглянуті теоретичні аспекти асинхронності та неблокуючих операцій, а також шляхи реалізації рішень, що використовують ці підходи, в Java екосистемі.

З метою забезпечення високої якості тестування була написана окрема програма на Gatling, яка імітує одночасні виклики API тестового застосунку.

За результатами тестування було доведено:

- неефективність використання асинхронного підходу для однієї операції та для дуже простих операцій;
- вищу ефективність та надійність асинхронності при роботі з декількома важкими операціями;

					ІА61.260БАК.003 ПЗ	Арк.
						76
Змн.	Арк.	№ докум.	Підпис	Дата		

- необхідність використання правильних підходів при масштабуванні асинхронних додатків.

Розроблений проєкт вирішує проблему з I/O блокуванням та прискіає витоків пам'яті.

В напрямку роботи дипломного проєкту можна й потрібно продовжувати роботу над:

- підтримкою рідковживаних операцій;
- додаванням нових інтеграцій з іншими базами даних;
- дослідженням переваг використання асинхронного клієнту на неблокуючих операціях разом з іншими AWS ресурсами.

					ІА61.260БАК.003 ПЗ	Арк.
						77
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Spring Data JPA [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020) : <https://docs.spring.io/spring-data/jpa/docs/current/reference/html/#reference>
2. Boostchicken/spring-data-dynamodb [Репозиторій] – Режим доступу до ресурсу (дата звернення 12.05.2020): <https://github.com/boostchicken/spring-data-dynamodb>
3. Рейтинг мов програмування 2020 [Електронний ресурс] – Режим доступу до ресурсу (дата звернення 12.05.2020): <https://dou.ua/lenta/articles/language-rating-jan-2020/>
4. Effective Java 3rd edition / Best practices for the Java platform / Joshua Bloch / Addison Wesley Professional 2017.
5. Spring MVC [Електронний ресурс] – Режим доступу до ресурсу (12.05.2020): <https://habr.com/ru/post/336816/>
6. Building DynamoDB brick by brick [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://medium.com/swlh/building-dynamodb-brick-by-brick-237e0008b698>
7. Amazon DynamoDB: How it works [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/HowItWorks.html>
8. Partitioning Behavior of DynamoDB [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://dzone.com/articles/partitioning-behavior-of-dynamodb>
9. Docker [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://www.docker.com/>
10. Apache Maven [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <http://maven.apache.org/guides/index.html>
11. Apache Tomcat [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <http://tomcat.apache.org/>

					ІА61.260БАК.003 ПЗ	Арк.
						78
Змн.	Арк.	№ докум.	Підпис	Дата		

12. The Scala programming language [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://www.scala-lang.org/>
13. Веб документація MDN [Електронна бібліотека] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://developer.mozilla.org/ru/docs/Web/JavaScript>
14. Gatling documentation [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://gatling.io/docs/current/>
15. Pub/Sub Architecture [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://cloud.google.com/pubsub/architecture>
16. «Дизайн-патерни — просто, як двері» / Будай Андрій Іванович, 2012.
17. Design Patterns Elements of reusable Object-Oriented Software / Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides / Addison Wesley Professional 1994.
18. Amazon DynamoDB Pricing [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://aws.amazon.com/dynamodb/pricing/>
19. Amazon VPC Pricing [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://aws.amazon.com/vpc/pricing/>
20. Amazon EC2 Pricing [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://aws.amazon.com/ec2/pricing/>
21. Java 8 Completable Future Частина 1 [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://vertex-academy.com/tutorials/ru/java-8-completablefuture/>
22. Java 8 Completable Future Частина 2 [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://vertex-academy.com/tutorials/ru/java-8-completablefuture-part-2/>
23. How does non-blocking IO work under the hood? [Електронний ресурс] – Режим доступу до ресурсу (дата звернення: 10.05.2020): <https://medium.com/ing-blog/how-does-non-blocking-io-work-under-the-hood-6299d2953c74>

					ІА61.260БАК.003 ПЗ	Арк.
						79
Змн.	Арк.	№ докум.	Підпис	Дата		

ДОДАТОК А

ЛІСТИНГ КОДУ БІБЛІОТЕКИ ТА КЛІЄНТСЬКОГО ТЕСТОВОГО ДОДАТКУ

```
package com.tiurINVALery.springdata.sdk2.annotations;
```

```
import java.lang.annotation.ElementType;  
import java.lang.annotation.Retention;  
import java.lang.annotation.RetentionPolicy;  
import java.lang.annotation.Target;
```

```
@Retention(RetentionPolicy.RUNTIME)  
@Target(ElementType.FIELD)  
public @interface Attribute {
```

```
    String name();  
}
```

```
package com.tiurINVALery.springdata.sdk2.annotations;
```

```
import java.lang.annotation.ElementType;  
import java.lang.annotation.Retention;  
import java.lang.annotation.RetentionPolicy;  
import java.lang.annotation.Target;
```

```
@Target({ ElementType.TYPE })  
@Retention(RetentionPolicy.RUNTIME)
```

```
public @interface CrudRepository {
```

```
    Class clazz();  
}
```

```
package com.tiurINVALery.springdata.sdk2.annotations;
```

```
import java.lang.annotation.ElementType;  
import java.lang.annotation.Retention;  
import java.lang.annotation.RetentionPolicy;  
import java.lang.annotation.Target;
```

```
@Retention(RetentionPolicy.RUNTIME)  
@Target(ElementType.FIELD)  
public @interface DynamoDB2IndexHashKey {  
    String attributeName() default "";  
    String globalSecondaryIndexName() default "";  
}
```

```
package com.tiurINVALery.springdata.sdk2.annotations;
```

```

import software.amazon.awssdk.services.dynamodb.model.KeyType;

import java.lang.annotation.ElementType;
import java.lang.annotation.Retention;
import java.lang.annotation.RetentionPolicy;
import java.lang.annotation.Target;

@Retention(RetentionPolicy.RUNTIME)
@Target(ElementType.FIELD)
public @interface Key {

    String fieldName();

    KeyType keyType();
}
package com.tiurinalery.springdata.sdk2.annotations;

import java.lang.annotation.ElementType;
import java.lang.annotation.Retention;
import java.lang.annotation.RetentionPolicy;
import java.lang.annotation.Target;

@Retention(RetentionPolicy.RUNTIME)
@Target(ElementType.TYPE)
public @interface Table {

    String tableName();

    Class clazz();
}
package com.tiurinalery.springdata.sdk2.entities;

import lombok.AllArgsConstructor;
import lombok.Builder;
import lombok.Data;
import lombok.NoArgsConstructor;

@Data
@AllArgsConstructor
@NoArgsConstructor
@Builder
public class Index {
    private String fieldName;
    private String dbAttributeName;
    private String secondaryIndexName;
}
package com.tiurinalery.springdata.sdk2.entities;

import com.tiurinalery.springdata.sdk2.parser.data.KeyProperties;
import lombok.AllArgsConstructor;
import lombok.Builder;
import lombok.Data;

```

```

import lombok.NoArgsConstructor;

import java.util.ArrayList;
import java.util.List;
import java.util.Map;

@Data
@AllArgsConstructor
@NoArgsConstructor
@Builder
public class Item {
    private Class clazz;
    private Map<String, String> codeAndDbFieldMapping;
    private Map<String, KeyProperties> keys;
    private String tableName;
    @Builder.Default
    private List<Index> indexes = new ArrayList<>();
}
package com.tiurinalery.springdata.sdk2.parser.data;

import lombok.AllArgsConstructor;
import lombok.Builder;
import lombok.Data;
import lombok.NoArgsConstructor;
import software.amazon.awssdk.services.dynamodb.model.KeyType;

@Data
@AllArgsConstructor
@NoArgsConstructor
@Builder
public class KeyProperties implements Parseable {

    private String fieldName;
    private KeyType keyType;
}
package com.tiurinalery.springdata.sdk2.parser.data;

public interface Parseable {
}
package com.tiurinalery.springdata.sdk2.parser;

import com.tiurinalery.springdata.sdk2.annotations.Attribute;
import com.tiurinalery.springdata.sdk2.annotations.DynamoDB2IndexHashKey;
import com.tiurinalery.springdata.sdk2.annotations.Key;
import com.tiurinalery.springdata.sdk2.annotations.Table;
import com.tiurinalery.springdata.sdk2.entities.Index;
import com.tiurinalery.springdata.sdk2.entities.Item;
import com.tiurinalery.springdata.sdk2.parser.data.KeyProperties;

import java.lang.reflect.Field;
import java.util.HashMap;
import java.util.List;

```

```

import java.util.Map;

public class EntityParser {

    private EntityParser() {

    }

    public static void defineKeyAndAttribute(List<Item> itemList, Field[] fields, Item
item) {
        Map<String, String> fieldMappingMap = new HashMap<>();
        Key key = null;
        for (Field f : fields) {
            key = parseField(item, fieldMappingMap, key, f);
        }
        item.setCodeAndDbFieldMapping(fieldMappingMap);
        itemList.add(item);
    }

    public static Key parseField(Item item, Map<String, String> fieldMappingMap, Key
key, Field f) {
        key = parseKey(item, key, f);
        parseAttributes(fieldMappingMap, f);
        defineIndexForSearching(f, item);
        return key;
    }

    protected static Key parseKey(Item item, Key key, Field f) {
        if (null == key) {
            Key[] annotationsByType = f.getAnnotationsByType(Key.class);
            if (annotationsByType.length == 1) {
                key = annotationsByType[0];
                item.setKeys(Map.of(f.getName(),
KeyProperties.builder().fieldName(key.fieldName()).keyType(key.keyType()).build()));
            }
            } else {
                // throw new RuntimeException("Support for more than 1 Key per entity not
implemented yet");
            }
            return key;
        }

    protected static void parseAttributes(Map<String, String> fieldMappingMap, Field f)
{
        try {
            Attribute[] annotationsByType = f.getAnnotationsByType(Attribute.class);
            if (annotationsByType != null && annotationsByType.length > 0) {
                fieldMappingMap.put(f.getName(), annotationsByType[0].name());
            }

        } catch (NullPointerException npe) {

```

```

        System.out.println(npe);
    }
}

public static void defineTableName(Class c, Item item) {
    item.setClazz(c);
    try {
        Table table = (Table) c.getAnnotationsByType(Table.class)[0];
        item.setTableName(table.tableName());
    } catch (NullPointerException npe) {
        System.out.println(npe);
    }
}

public static void defineIndexForSearching(Field f, Item item) {
    DynamoDB2IndexHashKey[] annotationsByType =
f.getAnnotationsByType(DynamoDB2IndexHashKey.class);
    if (null != annotationsByType && annotationsByType.length > 0) {
        DynamoDB2IndexHashKey key = annotationsByType[0];
        Index index = Index.builder()
            .fieldName(f.getName())
            .dbAttributeName(key.attributeName())
            .secondaryIndexName(key.globalSecondaryIndexName())
            .build();
        item.getIndexes().add(index);
    }
}
}

package com.tiurINVALery.springdata.sdk2.repository.impl;

import com.tiurINVALery.springdata.sdk2.entities.Index;
import com.tiurINVALery.springdata.sdk2.entities.Item;
import com.tiurINVALery.springdata.sdk2.repository.DynamoDbCrudRepo;
import com.tiurINVALery.springdata.sdk2.service.TargetComponentsLoaderServiceImpl;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Component;
import software.amazon.awssdk.services.dynamodb.DynamoDbAsyncClient;
import software.amazon.awssdk.services.dynamodb.model.AttributeValue;
import software.amazon.awssdk.services.dynamodb.model.ComparisonOperator;
import software.amazon.awssdk.services.dynamodb.model.Condition;
import software.amazon.awssdk.services.dynamodb.model.DeleteItemRequest;
import software.amazon.awssdk.services.dynamodb.model.DeleteItemResponse;
import software.amazon.awssdk.services.dynamodb.model.GetItemRequest;
import software.amazon.awssdk.services.dynamodb.model.GetItemResponse;
import software.amazon.awssdk.services.dynamodb.model.PutItemRequest;
import software.amazon.awssdk.services.dynamodb.model.PutItemResponse;
import software.amazon.awssdk.services.dynamodb.model.QueryRequest;
import software.amazon.awssdk.services.dynamodb.model.QueryResponse;

import javax.validation.Valid;
import javax.validation.constraints.NotNull;
import java.lang.reflect.Field;

```



```

import java.util.ArrayList;
import java.util.Collection;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.concurrent.CompletableFuture;
import java.util.stream.Collectors;

@Component
public class DynamoDbCrudRepoImpl implements DynamoDbCrudRepo {

    @Autowired
    private DynamoDbAsyncClient dynamoDbAsyncClient;

    @Autowired
    private TargetComponentsLoaderServiceImpl classLoaderService;

    @Override
    public CompletableFuture<PutItemResponse> save(Object object) {
        Item targetItem = classLoaderService.getEntityItems().stream().filter(item ->
item.getClass().isInstance(object)).findFirst().orElse(null);

        if (null != targetItem) {

            List<Map<String, String>> collect =
targetItem.getCodeAndDbFieldMapping().entrySet().stream()
                .map(es -> {
                    try {
                        Field field = targetItem.getClass().getDeclaredField(es.getKey());
                        field.setAccessible(true);
                        Object value = field.get(object);
                        if (null != value) {
                            return Map.of(es.getValue(), value.toString());
                        } else {
                            return Map.of(es.getValue(), "NOLL");
                        }
                    } catch (NoSuchFieldException fieldException) {
                        throw new RuntimeException("Error with field mapping");
                    } catch (IllegalAccessException illegalAccess) {
                        throw new RuntimeException("Trouble with field access");
                    }
                })
                .filter(this::deleteEmptyValues)
                .collect(Collectors.toList());

            Map<String, AttributeValue> requestMap = new HashMap<>();
            for (Map<String, String> map : collect) {
                map.forEach((key, value) -> requestMap.put(key,
AttributeValue.builder().s(value).build()));
            }

            PutItemRequest itemRequest = PutItemRequest.builder()
                .tableName(targetItem.getTableName())
                .item(requestMap)

```

```

        .build();

        return dynamoDbAsyncClient.putItem(itemRequest);
    } else {
        throw new RuntimeException("Unknown entity");
    }
}

@Override
public CompletableFuture<GetItemResponse> findById(Object objectWithIds) {
    Item targetItem = classLoaderService.getEntityItems().stream().filter(item ->
item.getClass().isInstance(objectWithIds)).findFirst().orElse(null);
    if (null != targetItem) {
        Map<String, AttributeValue> keyToGet = new HashMap<>();
        targetItem.getKeys().forEach((key, value) -> {
            try {
                Field declaredField = objectWithIds.getClass().getDeclaredField(key);
                declaredField.setAccessible(true);
                keyToGet.put(value.getFieldName(),
AttributeValue.builder().s(declaredField.get(objectWithIds).toString()).build());
            } catch (NoSuchFieldException noFieldException) {
                throw new RuntimeException("Problem with field pasrsing on get");
            } catch (IllegalAccessException ex) {
                throw new RuntimeException("Can't access field on get");
            }
        });
        CompletableFuture<GetItemResponse> itemsFuture =
dynamoDbAsyncClient.getItem(GetItemRequest.builder()
            .tableName(targetItem.getTableName())
            .key(keyToGet)
            .build());
        return itemsFuture;
    }
    throw new RuntimeException("Class for getItem not found");
}

@Override
public CompletableFuture<QueryResponse> findBySecondaryIndexes(@Valid
@NotNull Object objectWithIndex) {
    Item item = classLoaderService.getEntityItems().stream().filter(item1 ->
item1.getClass().isInstance(objectWithIndex)).findFirst().orElse(null);
    if (null != item) {
        List<Index> indexes = item.getIndexes();

        String secondaryIndexName = null;
        Map<String, List<AttributeValue>> dbAttributeNameToValRestrictions = new
HashMap<>();

        for (int i = 0; i < indexes.size(); i++) {
            secondaryIndexName = processIndex(objectWithIndex, indexes,
secondaryIndexName, dbAttributeNameToValRestrictions, i);
        }
    }
}

```

```

Map<String, Condition> conditions = new HashMap<>();

dbAttributeNameToValRestrictions.forEach((key, value) -> {

    conditions.put(key, Condition.builder()
        .attributeValueList(value)
        .comparisonOperator(ComparisonOperator.EQ)
        .build());
});

QueryRequest queryRequest = QueryRequest.builder()
    .tableName(item.getTableName())
    .indexName(secondaryIndexName)
    .keyConditions.conditions)
    .build();

return dynamoDbAsyncClient.query(queryRequest);
}
throw new RuntimeException("Can't find metadata for provided object, looks like it
not in provided for scan package.");
}

@Override
public CompletableFuture<DeleteItemResponse> delete(Object objectWithIds) {
    Item targetItem = classLoaderService.getEntityItems().stream().filter(item ->
item.getClass().isInstance(objectWithIds)).findFirst().orElse(null);
    if (null != targetItem) {
        Map<String, AttributeValue> keyToDelete = new HashMap<>();
        targetItem.getKeys().forEach((key, value) -> {
            try {
                Field declaredField = objectWithIds.getClass().getDeclaredField(key);
                declaredField.setAccessible(true);
                keyToDelete.put(value.getFieldName(),
AttributeValue.builder().s(declaredField.get(objectWithIds).toString()).build());
            } catch (NoSuchFieldException noFieldException) {
                throw new RuntimeException("Problem with field parsing on get");
            } catch (IllegalAccessException ex) {
                throw new RuntimeException("Can't access field on get");
            }
        });
        return dynamoDbAsyncClient.deleteItem(DeleteItemRequest.builder()
            .tableName(targetItem.getTableName())
            .key(keyToDelete)
            .build());
    }
    throw new RuntimeException("Class for getItem not found");
}

private boolean deleteEmptyValues(Map<String, String> originMap) {
    for (Map.Entry entry : originMap.entrySet()) {
        if (!"NOLL".equals(entry.getValue())) {

```

```

        return true;
    }
}
return false;
}

protected String processIndex(Object objectWithIndex, @NotNull List<Index>
indexes, String secondaryIndexName, Map<String, List<AttributeValue>>
dbAttributeNameToValRestrictions, int i) {
    Index index = indexes.get(i);
    try {
        Object objVal = getFieldValue(objectWithIndex, index);
        if (null != objVal) {
            secondaryIndexName =
checkThatAllKeysForSameIndex(secondaryIndexName, index);
            if (objVal instanceof Collection) {
                List<AttributeValue> vals = parseCollectionValues((List) objVal);
                dbAttributeNameToValRestrictions.put(index.getDbAttributeName(), vals);
            } else {
                dbAttributeNameToValRestrictions.put(index.getDbAttributeName(),
List.of(AttributeValue.builder()
                .s((String) objVal)
                .build()));
            }
        }
    } catch (NoSuchFieldException | IllegalAccessException e) {
        throw new RuntimeException("Troubles with field parsing via
findBySecondaryIndexes");
    }
    return secondaryIndexName;
}

```

```

protected String checkThatAllKeysForSameIndex(String secondaryIndexName, Index
index) {
    String secondaryIndexNameNext = index.getSecondaryIndexName();
    if (secondaryIndexName == null) {
        secondaryIndexName = secondaryIndexNameNext;
    } else if (secondaryIndexName != null &&
!secondaryIndexName.equals(secondaryIndexNameNext)) {
        throw new RuntimeException("This is impossible to search base on more than 1
index at the same time");
    }
    return secondaryIndexName;
}

```

```

protected Object getFieldValue(Object objectWithIndex, Index index) throws
NoSuchFieldException, IllegalAccessException {
    Field declaredField =
objectWithIndex.getClass().getDeclaredField(index.getFieldName());
    declaredField.setAccessible(true);
    return declaredField.get(objectWithIndex);
}

```

```

    }

    protected List<Attribute Value> parseCollectionValues(List objVal) {
        List<Attribute Value> vals = (List<Attribute Value>) objVal.stream().map(v ->
Attribute Value.builder()
                .s((String) v)
                .build()).collect(Collectors.toList());
        return vals;
    }
}package com.tiurinalery.springdata.sdk2.repository;

import software.amazon.awssdk.services.dynamodb.model.DeleteItemResponse;
import software.amazon.awssdk.services.dynamodb.model.GetItemResponse;
import software.amazon.awssdk.services.dynamodb.model.PutItemResponse;
import software.amazon.awssdk.services.dynamodb.model.QueryResponse;

import java.util.concurrent.CompletableFuture;

public interface DynamoDbCrudRepo {
    CompletableFuture<PutItemResponse> save(Object object);

    CompletableFuture<GetItemResponse> findById(Object objectWithIds);

    CompletableFuture<DeleteItemResponse> delete(Object objectWithIds);

    CompletableFuture<QueryResponse> findBySecondaryIndexes(Object
objectWithIndex);
}

package com.tiurinalery.springdata.sdk2.service;

import com.tiurinalery.springdata.sdk2.annotations.Table;
import com.tiurinalery.springdata.sdk2.entities.Item;
import org.springframework.beans.factory.annotation.AnnotatedBeanDefinition;
import org.springframework.beans.factory.annotation.Value;
import org.springframework.beans.factory.config.BeanDefinition;
import
org.springframework.context.annotation.ClassPathScanningCandidateComponentProvider;
import org.springframework.core.type.filter.AnnotationTypeFilter;
import org.springframework.stereotype.Service;

import javax.annotation.PostConstruct;
import java.lang.reflect.Field;
import java.util.HashSet;
import java.util.LinkedList;
import java.util.List;
import java.util.Set;

import static
com.tiurinalery.springdata.sdk2.parser.EntityParser.defineKeyAndAttribute;
import static com.tiurinalery.springdata.sdk2.parser.EntityParser.defineTableName;

```

```

@Service
public class TargetComponentsLoaderServiceImpl {

    @Value("${spring.data.dynamodb.sdkv2.packages.entities}")
    private String entities;

    private Set<Class> entityClasses;

    private List<Item> entityItems;

    public List<Item> getEntityItems() {
        return entityItems;
    }

    @PostConstruct
    public void init() {
        createClassesSet();
        createItemList();
    }

    public void createClassesSet() {
        entityClasses = new HashSet<>();

        //Create entity classes set
        ClassPathScanningCandidateComponentProvider scanner = new
ClassPathScanningCandidateComponentProvider(false) {
            @Override
            protected boolean isCandidateComponent(AnnotatedBeanDefinition
beanDefinition) {
                return super.isCandidateComponent(beanDefinition) ||
beanDefinition.getMetadata().isAbstract() || beanDefinition.getMetadata().isInterface();
            }
        };
        scanner.addIncludeFilter(new AnnotationTypeFilter(Table.class));

        for (BeanDefinition beanDefinition : scanner
            .findCandidateComponents(entities)) {
            try {
                entityClasses.add(Class.forName(beanDefinition.getBeanClassName()));
            } catch (ClassNotFoundException cnfe) {
                System.out.println(cnfe);
            }
        }
    }

    void createItemList() {
        final List<Item> itemList = new LinkedList<>();
        for (Class c : entityClasses) {
            Field[] fields = c.getDeclaredFields();

```

```

        Item item = new Item();

        defineTableName(c, item);
        defineKeyAndAttribute(itemList, fields, item);
    }

    entityItems = itemList;
}

}

package com.tiurINVALERY.springdata.sdk2;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

//TODO: It must use just for testing. Deprecate in future.
@SpringBootApplication
public class DynamoSDK2 {

    public static void main(String[] arg){
        SpringApplication.run(DynamoSDK2.class,arg);
    }
}package com.tiurINVALERY.springdata.sdk2;

import org.springframework.context.annotation.ComponentScan;
import org.springframework.context.annotation.Configuration;

@Configuration
@ComponentScan(value = {"com.tiurINVALERY.springdata.sdk2.repository",
"com.tiurINVALERY.springdata.sdk2.parser",
    "com.tiurINVALERY.springdata.sdk2.service"})
public class StartDynamoDb2 {
}
package springdata.sdk2.example.app.config;

import org.springframework.beans.factory.annotation.Value;
import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import software.amazon.awssdk.auth.credentials.AwsBasicCredentials;
import software.amazon.awssdk.auth.credentials.AwsCredentialsProvider;
import software.amazon.awssdk.auth.credentials.StaticCredentialsProvider;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.services.dynamodb.DynamoDbAsyncClient;
import software.amazon.awssdk.services.dynamodb.DynamoDbClient;

import java.net.URI;

@Configuration
public class AbstractDynamoDBConfig {

    @Value("qwerty")
    private String amazonAWSAccessKey;

```

```

@Value("${amazon.aws.secretkey}")
private String amazonAWSSecretKey;

@Value("${amazon.aws.signing.region:us-west-2}")
private String amazonSigningRegion;

@Value("${amazon.aws.dynamodb.endpoint:http://localhost:8000}")
private String amazonDynamoEndpoint;

@Bean
public DynamoDbAsyncClient dynamoDbAsyncClient() {

    return DynamoDbAsyncClient.builder()
        .endpointOverride(URI.create("http://localhost:8000"))
        .credentialsProvider(awsCredentialsProvider())
        .region(Region.US_WEST_2)
        .build();
}

@Bean
public AwsCredentialsProvider awsCredentialsProvider(){
    return StaticCredentialsProvider.create(
        AwsBasicCredentials.create(amazonAWSAccessKey,amazonAWSSecretKey));
}

@Bean
public DynamoDbClient dynamoDbClient() {
    return DynamoDbClient.builder()
        .endpointOverride(URI.create(amazonDynamoEndpoint))
        .credentialsProvider(awsCredentialsProvider())
        .region(Region.US_WEST_2)
        .build();
}
}package springdata.sdk2.example.app.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestParam;
import org.springframework.web.bind.annotation.RestController;
import software.amazon.awssdk.services.dynamodb.model.AttributeValue;
import software.amazon.awssdk.services.dynamodb.model.GetItemResponse;
import software.amazon.awssdk.services.dynamodb.model.PutItemResponse;
import software.amazon.awssdk.services.dynamodb.model.QueryResponse;

```



```

import springdata.sdk2.example.app.model.User;
import springdata.sdk2.example.app.service.GenerateUsersForLoadTestService;
import springdata.sdk2.example.app.service.UserServiceImpl;
import springdata.sdk2.example.app.service.UserServiceSyncImpl;

import java.util.List;
import java.util.Map;
import java.util.Objects;
import java.util.concurrent.CompletableFuture;
import java.util.concurrent.ConcurrentHashMap;
import java.util.concurrent.ExecutionException;
import java.util.concurrent.TimeUnit;
import java.util.concurrent.TimeoutException;

@RestController
public class UserController {

    private ConcurrentHashMap<String, User> users = new ConcurrentHashMap<>();

    @Autowired
    private UserServiceImpl userService;
    @Autowired
    private UserServiceSyncImpl userServiceSync;
    @Autowired
    private GenerateUsersForLoadTestService generateUsersForLoadTestService;

    @GetMapping("/user/{user_id}")
    public ResponseEntity<Long> getUser(@PathVariable(name = "user_id") String
userId) throws InterruptedException, ExecutionException {
        long start = System.currentTimeMillis();
        userService.getUser(userId).get();
        long end = System.currentTimeMillis();
        return ResponseEntity.ok(end - start);
    }

    @GetMapping("/user/sync/{user_id}")
    public ResponseEntity<Long> getUserSync(@PathVariable(name = "user_id") String
userId) {
        long start = System.currentTimeMillis();
        Map<String, String> uuid = Map.of("UUID", userId);
        userServiceSync.getUserSync("PROD_USER", uuid);
        long end = System.currentTimeMillis();
        return ResponseEntity.ok(end - start);
    }

    @GetMapping("/user/username/{username}")
    public ResponseEntity<QueryResponse> getUserViaUsername(@PathVariable(name
= "username") String username) {
        return ResponseEntity.ok(userService.findUserByUsername(username).join());
    }
}

```

```

    @PostMapping("/user/create")
    public ResponseEntity<Boolean> createUser(@RequestBody User user) {
        return
ResponseEntity.ok(userService.saveUser(user).join().sdkHttpResponse().isSuccessful());
    }

    @GetMapping("/user/generate")
    public ResponseEntity<List<User>>
generateUsersForLoadTest(@RequestParam("capacity") Integer capacity) {
        return ResponseEntity.ok(generateUsersForLoadTestService.getUsers(capacity));
    }

    @PostMapping("/user/sync/create")
    public ResponseEntity<Boolean> createUserSync(@RequestBody User user) {
        Map<String, AttributeValue> userAttributes = Map.of("UUID",
AttributeValue.builder().s(user.getUuid()).build(),
"USERNAME", AttributeValue.builder().s(user.getUsername()).build());

        return ResponseEntity.ok(userServiceSync.saveUserSyncClient(
"PROD_USER", userAttributes).sdkHttpResponse().isSuccessful());
    }

    @PostMapping("/user/fullsync")
    public ResponseEntity<Long> startStressFLoW(@RequestBody User user) throws
InterruptedException, ExecutionException, TimeoutException {
        Long startTime = System.currentTimeMillis();
        CompletableFuture<GetItemResponse> checkThatUserAbsent =
userService.getUser(user.getUuid());
        checkThatUserAbsent
            .thenApplyAsync(fn -> fn.item().get("USERNAME").s().isEmpty());
        checkThatUserAbsent.get(1200, TimeUnit.MILLISECONDS);

        CompletableFuture<PutItemResponse> putItemResponseCompletableFuture =
userService.saveUser(user);

        putItemResponseCompletableFuture
            .thenAcceptAsync(fn -> {
                try {
                    userService.getUser(user.getUuid()).get(1200,
TimeUnit.MILLISECONDS);
                } catch (InterruptedException | ExecutionException | TimeoutException e) {
                }
            })
            .return();

        Long endTime = System.currentTimeMillis();
        return ResponseEntity.ok(endTime - startTime);
    }

    @PostMapping("/user/sync/fullsync")

```

```

public ResponseEntity<Long> startStressFlowSync(@RequestBody User user) {
    Long startTime = System.currentTimeMillis();

    Map<String, String> uuid = Map.of("UUID", user.getUuid());
    GetItemResponse prodUser = userServiceSync.getUserSync("PROD_USER",
uuid);

    if (prodUser.getValueForField("USERNAME", String.class).isEmpty()) {

        Map<String, AttributeValue> userAttributes = Map.of("UUID",
AttributeValue.builder().s(user.getUuid()).build(),
"USERNAME", AttributeValue.builder().s(user.getUsername()).build());

        userServiceSync.saveUserSyncClient("PROD_USER", userAttributes);

        userServiceSync.getUserSync("PROD_USER", uuid);

        Long endTime = System.currentTimeMillis();
        return ResponseEntity.ok(endTime - startTime);
    }
    return ResponseEntity.ok(null);
}

@PostMapping("/user/create/operations")
public ResponseEntity<Long> asyncCreateAndLogic(@RequestBody User user)
throws InterruptedException, ExecutionException, TimeoutException {
    Long startTime = System.currentTimeMillis();
    CompletableFuture<PutItemResponse> putItemResponseCompletableFuture =
userService.saveUser(user);
    putItemResponseCompletableFuture
        .thenApplyAsync(fn -> users.putIfAbsent(user.getUsername(), user))
        .thenApplyAsync(Objects::nonNull)
        .thenApply(fn -> user.getUsername().hashCode());
    putItemResponseCompletableFuture.get(400, TimeUnit.MILLISECONDS);
    Long endTime = System.currentTimeMillis();
    return ResponseEntity.ok(endTime - startTime);
}

@PostMapping("/user/sync/create/operations")
public ResponseEntity<Long> syncCreateAndLogic(@RequestBody User user) {
    Long startTime = System.currentTimeMillis();
    Map<String, AttributeValue> userAttributes = Map.of("UUID",
AttributeValue.builder().s(user.getUuid()).build(),
"USERNAME", AttributeValue.builder().s(user.getUsername()).build());

    userServiceSync.saveUserSyncClient("PROD_USER", userAttributes);

    User existUser = users.putIfAbsent(user.getUsername(), user);
    if (null != existUser) {
        user.getUsername().hashCode();
    }
    Long endTime = System.currentTimeMillis();

```

```

        if ((endTime - startTime) < 400L) {
            return ResponseEntity.ok(endTime - startTime);
        } else {
            return new ResponseEntity<>(endTime - startTime,
HttpStatus.INTERNAL_SERVER_ERROR);
        }
    }
}

package springdata.sdk2.example.app.exception;

public class InvalidRequiredAttributeException extends RuntimeException {
}

package springdata.sdk2.example.app.model;

import com.tiurinalery.springdata.sdk2.annotations.Attribute;
import com.tiurinalery.springdata.sdk2.annotations.DynamoDB2IndexHashKey;
import com.tiurinalery.springdata.sdk2.annotations.Key;
import com.tiurinalery.springdata.sdk2.annotations.Table;
import lombok.AllArgsConstructor;
import lombok.Builder;
import lombok.Data;
import lombok.NoArgsConstructor;
import software.amazon.awssdk.services.dynamodb.model.KeyType;

@Table(tableName = "PROD_USER", clazz = User.class)
@Data
@AllArgsConstructor
@NoArgsConstructor
@Builder
public class User {

    @Key(fieldName = "UUID", keyType = KeyType.HASH)
    @Attribute(name = "UUID")
    private String uuid;

    @Attribute(name = "USERNAME")
    @DynamoDB2IndexHashKey(attributeName = "USERNAME",
globalSecondaryIndexName = "idndx_username")
    private String username;

    @Attribute(name = "APPROVED")
    private String approvedUser;
}

package springdata.sdk2.example.app.service;

import org.springframework.stereotype.Service;
import springdata.sdk2.example.app.model.User;

import java.util.ArrayList;
import java.util.List;

```

```

import java.util.Random;

@Service
public class GenerateUsersForLoadTestService {

    private Random random = new Random();

    public List<User> getUsers(Integer capacity) {
        List<User> users = new ArrayList<>(capacity);

        for (int i = 0; i < capacity; i++) {
            String uuid = "uuid-"+i+"_"+random.nextLong();
            String username = "username-"+i+"_"+random.nextLong();
            users.add(User.builder()
                .uuid(uuid)
                .username(username)
                .build());
        }
        return users;
    }
}

package springdata.sdk2.example.app.service;

import com.tiurinalery.springdata.sdk2.repository.DynamoDbCrudRepo;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import software.amazon.awssdk.services.dynamodb.model.DeleteItemResponse;
import software.amazon.awssdk.services.dynamodb.model.GetItemResponse;
import software.amazon.awssdk.services.dynamodb.model.PutItemResponse;
import software.amazon.awssdk.services.dynamodb.model.QueryResponse;
import springdata.sdk2.example.app.exception.InvalidRequiredAttributeException;
import springdata.sdk2.example.app.model.User;

import javax.annotation.PostConstruct;
import java.util.concurrent.CompletableFuture;

@Service
public class UserServiceImpl {

    @Autowired
    private DynamoDbCrudRepo dynamoDbCrudRepo;

    // @PostConstruct
    // public void test() {
    //     User user = User.builder()
    //         .username("username")
    //         .uuid("swa-123-dfsas")
    //         .build();
    //     dynamoDbCrudRepo.findById(user).join();
    //     dynamoDbCrudRepo.save(user).join();
    // }

```

```

//    dynamoDbCrudRepo.findById(user).join();
//    dynamoDbCrudRepo.findByIdBySecondaryIndexes(user).join();
//    dynamoDbCrudRepo.delete(user).join();
//    dynamoDbCrudRepo.findById(user).join();
// }

public CompletableFuture<GetItemResponse> getUser(String uuid) {
    return dynamoDbCrudRepo.findById(User.builder().uuid(uuid).build());
}

public CompletableFuture<PutItemResponse> saveUser(User user) {
    if (null == user || null == user.getUuid() || !user.getUuid().contains("uuid")) {
        throw new InvalidRequiredAttributeException();
    }
    return dynamoDbCrudRepo.save(user);
}

public DeleteItemResponse deleteUser(String uuid) {
    return dynamoDbCrudRepo.delete(User.builder().uuid(uuid).build()).join();
}

public CompletableFuture<QueryResponse> findUserByUsername(String username) {
    return
    dynamoDbCrudRepo.findByIdBySecondaryIndexes(User.builder().username(username).build());
}

public void approveUser(String uuid) {
    GetItemResponse userCreated = getUser(uuid).join();

    if (userCreated.item().size() > 0) {
        String username = userCreated.item().get("USERNAME").s();
        if (checkUserEmailApprovedFromThirdParty(username)) {
            saveUser(User.builder()
                .username(username)
                .uuid(uuid)
                .approvedUser("y")
                .build());
        } else {
            saveUser(User.builder()
                .username(username)
                .uuid(uuid)
                .approvedUser("n")
                .build());
        }
    }
}

private boolean checkUserEmailApprovedFromThirdParty(String username) {
    if (username.hashCode() % 2 == 0) {
        return true;
    }
}

```

```

        } else {
            return false;
        }
    }
}

package springdata.sdk2.example.app.service;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Service;
import software.amazon.awssdk.services.dynamodb.DynamoDbClient;
import software.amazon.awssdk.services.dynamodb.model.AttributeValue;
import software.amazon.awssdk.services.dynamodb.model.ComparisonOperator;
import software.amazon.awssdk.services.dynamodb.model.Condition;
import software.amazon.awssdk.services.dynamodb.model.GetItemRequest;
import software.amazon.awssdk.services.dynamodb.model.GetItemResponse;
import software.amazon.awssdk.services.dynamodb.model.PutItemRequest;
import software.amazon.awssdk.services.dynamodb.model.PutItemResponse;
import software.amazon.awssdk.services.dynamodb.model.QueryRequest;
import software.amazon.awssdk.services.dynamodb.model.QueryResponse;

import java.util.HashMap;
import java.util.List;
import java.util.Map;

@Service
public class UserServiceSyncImpl {

    @Autowired
    private DynamoDbClient dynamoDbClient;

    public PutItemResponse saveUserSyncClient(String tableName, Map<String,
AttributeValue> fields) {
        return dynamoDbClient.putItem(PutItemRequest.builder()
            .tableName(tableName)
            .item(fields)
            .build());
    }

    public GetItemResponse getUserSync(String tableName, Map<String, String> keys) {
        Map<String, AttributeValue> keyCollection = new HashMap<>();

        keys.forEach((key, value) -> keyCollection.put(key,
AttributeValue.builder().s(value).build()));

        return dynamoDbClient.getItem(GetItemRequest.builder()
            .tableName(tableName)
            .key(keyCollection)
            .build());
    }

    public QueryResponse findUserByUsername(String username) {
        Map<String, Condition> conditions = new HashMap<>();

```

```

        conditions.put("USERNAME", Condition.builder()
            .attributeValueList(List.of(AttributeValue.builder().s(username).build()))
            .comparisonOperator(ComparisonOperator.EQ)
            .build());

        return dynamoDbClient.query(QueryRequest.builder()
            .tableName("PROD_USER")
            .indexName("idndx_username")
            .keyConditions(conditions)
            .build());
    }
}

package springdata.sdk2.example.app;

import com.tiurinalery.springdata.sdk2.StartDynamoDb2;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.context.annotation.Import;

@SpringBootApplication
@Import(value = {StartDynamoDb2.class})
public class Application {

    public static void main(String [] args) {
        SpringApplication.run(Application.class);
    }
}

amazon.aws.secretkey = 5678
amazon.aws.accesskey = 1234
spring.data.dynamodb.sdkv2.schema.create= true
spring.data.dynamodb.sdkv2.packages.entities = springdata.sdk2.example.app.model
spring.data.dynamodb.sdkv2.packages.repositories = springdata.sdk2.example.app.repo

<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <parent>
        <artifactId>parent</artifactId>
        <groupId>com.tiurinalery.springdata.dynamodb.sdk2</groupId>
        <version>1.0-SNAPSHOT</version>
    </parent>
    <modelVersion>4.0.0</modelVersion>

    <artifactId>exampleapp</artifactId>
    <version>${parent.version}</version>

    <dependencies>
        <dependency>

```



```

        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter</artifactId>
        <version>2.2.4.RELEASE</version>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-autoconfigure</artifactId>
        <version>2.2.4.RELEASE</version>
    </dependency>

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
    <dependency>
        <groupId>com.tiurinalery.springdata.dynamodb.sdk2</groupId>
        <artifactId>core</artifactId>
        <version>1.0-SNAPSHOT</version>
    </dependency>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>
    </dependency>
</dependencies>

<build>
    <finalName>exampleapp</finalName>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-compiler-plugin</artifactId>
        </plugin>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-release-plugin</artifactId>
        </plugin>
        <plugin>
            <groupId>org.apache.maven.plugins</groupId>
            <artifactId>maven-javadoc-plugin</artifactId>
        </plugin>
    </plugins>
</build>

</project>

```

```

<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>com.tiurinalery.springdata.dynamodb.sdk2</groupId>
  <artifactId>parent</artifactId>
  <packaging>pom</packaging>
  <version>1.0-SNAPSHOT</version>

  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>2.2.4.RELEASE</version>
  </parent>

  <properties>
    <lombok.version>1.18.10</lombok.version>
    <spring-boot.version>2.2.4.RELEASE</spring-boot.version>
  </properties>

  <modules>
    <module>core</module>
    <module>example.app</module>
  </modules>

  <dependencies>
    <dependency>
      <groupId>org.projectlombok</groupId>
      <artifactId>lombok</artifactId>
      <version>${lombok.version}</version>
    </dependency>
  </dependencies>

  <dependencyManagement>
    <dependencies>
      <dependency>
        <groupId>software.amazon.awssdk</groupId>
        <artifactId>dynamodb</artifactId>
        <version>2.10.24</version>
      </dependency>
      <dependency>
        <groupId>com.tiurinalery.springdata.dynamodb.sdk2</groupId>
        <artifactId>core</artifactId>
        <version>${project.version}</version>
      </dependency>

      <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>

```

```

        <version>${spring-boot.version}</version>
        <scope>test</scope>
    </dependency>
</dependencies>
</dependencyManagement>

<build>
    <pluginManagement>
        <plugins>
            <plugin>
                <groupId>org.springframework.boot</groupId>
                <artifactId>spring-boot-maven-plugin</artifactId>
                <executions>
                    <execution>
                        <goals>
                            <goal>repackage</goal>
                        </goals>
                    </execution>
                </executions>
            </plugin>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-compiler-plugin</artifactId>
                <configuration>
                    <source>11</source>
                    <target>11</target>
                    <compilerArgument>-Xlint:all</compilerArgument>
                    <showWarnings>true</showWarnings>
                    <showDeprecation>true</showDeprecation>
                </configuration>
            </plugin>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-release-plugin</artifactId>
            </plugin>
            <plugin>
                <groupId>org.apache.maven.plugins</groupId>
                <artifactId>maven-javadoc-plugin</artifactId>
                <configuration>
                    <source>11</source>
                    <!-- Only show errors and warnings -->
                    <quiet>true</quiet>
                    <!-- skip=true if you want to stop generation of javadocs -->
                    <skip>true</skip>
                </configuration>
            </plugin>
        </plugins>
    </pluginManagement>
</build>

</project>

```

ДОДАТОК Б

НАЛАШТУВАННЯ ЛОКАЛЬНОГО КОНТЕЙНЕРУ З БАЗОЮ ДАНИХ

version: '2'

services:

dynamodb:

environment:

- AWS_ACCESS_KEY_ID=qwerty
- AWS_SECRET_ACCESS_KEY=5678
- AWS_REGION=us-west-2

image: amazon/dynamodb-local

ports:

- 8000:8000

ДОДАТОК В

ЛІСТИНГ SCALA ЗАСТОСУНКУ

```
<configuration>

  <appender name="CONSOLE" class="ch.qos.logback.core.ConsoleAppender">
    <encoder>
      <pattern>%d{HH:mm:ss.SSS} [%thread] %-5level %logger{36} -
%msg%n</pattern>
    </encoder>
  </appender>

  <logger name="io.gatling" level="INFO"/>
  <logger name="io.gatling.http.ahc" level="TRACE"/>

  <root level="INFO">
    <appender-ref ref="CONSOLE"/>
  </root>
</configuration>

package gatling.test.example.simulation

import io.gatling.core.Predef.{StringBody, constantUsersPerSec, global, scenario, _}
import io.gatling.http.Predef.{http, status, _}

import scala.concurrent.duration._

class AsyncGetUser extends Simulation {
  val jsonFileUsersFeeder = jsonFile("users_for_load_tests.json").circular
  val httpConf = http.baseUrl("http://localhost:8080")
  val postUsers = scenario("Async get user calls")
    .feed(jsonFileUsersFeeder)
    .exec(http("root end point")
      .get("/user/${uuid}")
      .check(status.is(200))
    )
  setUp(postUsers.inject(
    constantUsersPerSec(PerfTestConfig.requestPerSecond) during (2.0 minutes))
    .protocols(httpConf)
    .assertions(
      global.responseTime.max.lt(200),
      global.responseTime.mean.lt(10),
      global.successfulRequests.percent.gt(97)
    )
  )
}
```

```
package gatling.test.example.simulation

import io.gatling.core.Predef.{StringBody, constantUsersPerSec, global, scenario, _}
```

```

import io.gatling.http.Predef.{http, status, _}

import scala.concurrent.duration._

case class User(uuid: String, username: String)

class AsyncUserCreateSimulation extends Simulation {
  val jsonFileUsersFeeder = jsonFile("users_for_load_tests.json").circular
  val httpConf = http.baseUrl("http://localhost:8080")
  val postUsers = scenario("Async User creation")
    .feed(jsonFileUsersFeeder)
    .exec(http("root end point")
      .post("/user/create")
      .header("Content-Type", "application/json")
      .body(StringBody(
        """{
          "uuid": "${uuid}",
          "username": "${username}"
        }""")
      ).asJson
      .check(status.is(200))
    )
  setUp(postUsers.inject(
    constantUsersPerSec(PerfTestConfig.requestPerSecond) during (2.0 minutes))
    .protocols(httpConf)
    .assertions(
      global.responseTime.max.lt(200),
      global.responseTime.mean.lt(10),
      global.successfulRequests.percent.gt(97)
    )
  )
}

package gatling.test.example.simulation

import gatling.test.example.simulation.SystemPropertiesUtil._

object PerfTestConfig {
  val baseUrl = getAsStringOrElse("baseUrl", "http://localhost:8080")
  val requestPerSecond = getAsDoubleOrElse("requestPerSecond", 300f)
  val durationMin = getAsDoubleOrElse("durationMin", 2.0)
  val meanResponseTimeMs = getAsIntOrElse("meanResponseTimeMs", 10)
  val maxResponseTimeMs = getAsIntOrElse("maxResponseTimeMs", 500)
}

package gatling.test.example.simulation

import gatling.test.example.simulation.PerfTestConfig.{baseUrl, durationMin,
maxResponseTimeMs, meanResponseTimeMs}
import io.gatling.core.Predef.{StringBody, constantUsersPerSec, global, scenario, _}
import io.gatling.http.Predef.{http, status, _}

```

```
import scala.concurrent.duration._
```

```
class ScenarioSimulation extends Simulation {  
  val jsonFileUsersFeeder = jsonFile("users_for_load_tests.json").random  
  val httpConf = http.baseUrl("http://localhost:8080")  
  val postUsers = scenario("Stress scenario async")  
    .feed(jsonFileUsersFeeder)  
    .exec(http("root end point"))  
    .post("/user/fullsycle")  
    .header("Content-Type", "application/json")  
    .body(StringBody(  
      """"{  
        "uuid": "${uuid}",  
        "username": "${username}"  
      }""")  
    ).asJson  
    .check(status.is(200))  
  )  
  setUp(postUsers.inject(  
    constantUsersPerSec(PerfTestConfig.requestPerSecond) during (1.0 minutes))  
    .protocols(httpConf)  
    .assertions(  
      global.responseTime.max.lt(1200),  
      global.responseTime.mean.lt(10),  
      global.successfulRequests.percent.gt(97)  
    )  
  )  
}
```

```
package gatling.test.example.simulation
```

```
import io.gatling.core.Predef.{StringBody, constantUsersPerSec, global, scenario, _}  
import io.gatling.http.Predef.{http, status, _}
```

```
import scala.concurrent.duration._
```

```
class ScenarioWithSingleDbCallSimulation extends Simulation {  
  val jsonFileUsersFeeder = jsonFile("users_for_load_tests.json").random  
  val httpConf = http.baseUrl("http://localhost:8080")  
  val postUsers = scenario("Scenario with single db call and hard CPU logic after async")  
    .feed(jsonFileUsersFeeder)  
    .exec(http("root end point"))  
    .post("/user/create/operations")  
    .header("Content-Type", "application/json")  
    .body(StringBody(  
      """"{  
        "uuid": "${uuid}",  
        "username": "${username}"  
      }""")  
    ).asJson  
    .check(status.is(200))  
  )  
}
```

```

    )
    setUp(postUsers.inject(
      constantUsersPerSec(PerfTestConfig.requestPerSecond) during (2.0 minutes))
      .protocols(httpConf))
      .assertions(
        global.responseTime.max.lt(1200),
        global.responseTime.mean.lt(10),
        global.successfulRequests.percent.gt(97)
      )
    )
  }

```

```
package gatling.test.example.simulation
```

```
import io.gatling.core.Predef.{constantUsersPerSec, global, scenario, _}
import io.gatling.http.Predef.{http, status, _}
```

```
import scala.concurrent.duration._
```

```
class SyncGetUser extends Simulation {
  val jsonFileUsersFeeder = jsonFile("users_for_load_tests.json").circular
  val httpConf = http.baseUrl("http://localhost:8080")
  val postUsers = scenario("Sync Get users call")
    .feed(jsonFileUsersFeeder)
    .exec(http("root end point")
      .get("/user/sync/${uuid}")
      .check(status.is(200))
    )
  setUp(postUsers.inject(
    constantUsersPerSec(PerfTestConfig.requestPerSecond) during (2.0 minutes))
    .protocols(httpConf))
    .assertions(
      global.responseTime.max.lt(200),
      global.responseTime.mean.lt(10),
      global.successfulRequests.percent.gt(97)
    )
  )
}

```

```
package gatling.test.example.simulation
```

```
import io.gatling.core.Predef.{StringBody, constantUsersPerSec, global, scenario, _}
import io.gatling.http.Predef.{http, status, _}
```

```
import scala.concurrent.duration._
```

```
class SyncScenarioSimulation extends Simulation {
  val jsonFileUsersFeeder = jsonFile("users_for_load_tests.json").random
  val httpConf = http.baseUrl("http://localhost:8080")
  val postUsers = scenario("Stress scenario sync")
    .feed(jsonFileUsersFeeder)
    .exec(http("root end point")
      .post("/user/sync/fullcycle")
    )
  )
}

```



```

        .header("Content-Type", "application/json")
        .body(StringBody(
            """{
            "uuid": "${uuid}",
            "username": "${username}"
            }"""
        )).asJson
        .check(status.is(200))
    )
    setUp(postUsers.inject(
        constantUsersPerSec(PerfTestConfig.requestPerSecond) during (1.0 minutes))
        .protocols(httpConf))
        .assertions(
            global.responseTime.max.lt(200),
            global.responseTime.mean.lt(10),
            global.successfulRequests.percent.gt(97)
        )
    }

package gatling.test.example.simulation

import io.gatling.core.Predef.{StringBody, constantUsersPerSec, global, scenario, _}
import io.gatling.http.Predef.{http, status, _}

import scala.concurrent.duration._

class SyncScenarioWithSingleDbCallSimulation extends Simulation {
    val jsonFileUsersFeeder = jsonFile("users_for_load_tests.json").random
    val httpConf = http.baseUrl("http://localhost:8080")
    val postUsers = scenario("Scenario with single db call and hard CPU logic after sync")
        .feed(jsonFileUsersFeeder)
        .exec(http("root end point")
            .post("/user/sync/create/operations")
            .header("Content-Type", "application/json")
            .body(StringBody(
                """{
                "uuid": "${uuid}",
                "username": "${username}"
                }"""
            )).asJson
            .check(status.is(200))
        )
    setUp(postUsers.inject(
        constantUsersPerSec(PerfTestConfig.requestPerSecond) during (2.0 minutes))
        .protocols(httpConf))
        .assertions(
            global.responseTime.max.lt(1200),
            global.responseTime.mean.lt(10),
            global.successfulRequests.percent.gt(97)
        )
    }
}

```

```

package gatling.test.example.simulation

import com.google.gson.Gson
import io.gatling.core.Predef.{constantUsersPerSec, global, scenario, _}
import io.gatling.http.Predef.{http, status, _}

import scala.concurrent.duration._

class SyncUserCreateSimulation extends Simulation {
  val jsonFileUsersFeeder = jsonFile("users_for_load_tests.json").circular
  val httpConf = http.baseUrl("http://localhost:8080")
  val r = scala.util.Random
  val gson = new Gson()
  val postUsers = scenario("User creation sync")
    .feed(jsonFileUsersFeeder)
    .exec(http("root end point")
      .post("/user/sync/create")
      .header("Content-Type", "application/json")
      .body(StringBody(
        """"{
          "uuid": "${uuid}",
          "username": "${username}"
        }""")
      ).asJson
      .check(status.is(200))
    )
  setUp(postUsers.inject(
    constantUsersPerSec(PerfTestConfig.requestPerSecond) during (2.0 minutes))
    .protocols(httpConf)
    .assertions(
      global.responseTime.max.lt(200),
      global.responseTime.mean.lt(10),
      global.successfulRequests.percent.gt(97)
    )
  )
}

```

```

package gatling.test.example.simulation

import scala.sys.SystemProperties

object SystemPropertiesUtil {
  val systemProperties = new SystemProperties

  def getAsIntOrElse(property: String, default: Int): Int = {
    systemProperties.getOrElse(property, default).toString.toInt
  }

  def getAsStringOrElse(property: String, default: String): String = {
    systemProperties.getOrElse(property, default)
  }
}

```

```

def getAsBooleanOrElse(property: String, default: Boolean): Boolean = {
  systemProperties.getOrElse(property, default).toString().toBoolean
}

def getAsDoubleOrElse(property: String, default: Double): Double = {
  systemProperties.getOrElse(property, default).toString().toDouble
}

}

package gatling.test.example.helper

import gatling.test.example.simulation.PerfTestConfig
import io.gatling.app.Gatling
import io.gatling.core.config.GatlingPropertiesBuilder

object Engine {
  def start(): Unit = {
    println(s"Target app base url : ${PerfTestConfig.baseUrl}")
    val props = new GatlingPropertiesBuilder
    props.resourcesDirectory(IDEPathHelper.dataDirectory.toString)
    props.resultsDirectory(IDEPathHelper.resultsDirectory.toString)
    props.binariesDirectory(IDEPathHelper.mavenBinariesDirectory.toString)
    props.simulationsDirectory(IDEPathHelper.mavenSourcesDirectory.toString)
    Gatling.fromMap(props.build)
  }

  def main(args: Array[String]) = Engine.start
}

```

ДОДАТОК Г

НАБІР ТЕСТОВИХ ВІРТУАЛЬНИХ КОРИСТУВАЧІВ

```
[{"uid":"uid-0_-3576343584576578794","username":"username-0_-  
2887410622835289692","approvedUser":null}, {"uid":"uid-1_-8282120569429396238","username":"username-  
1_6638854851494094151","approvedUser":null}, {"uid":"uid-2_-5857215553080337704","username":"username-2_-  
2492945954316710694","approvedUser":null}, {"uid":"uid-3_-252892086489774405","username":"username-  
3_5621016358535253080","approvedUser":null}, {"uid":"uid-4_-4422694918774547790","username":"username-  
4_7881692771459506575","approvedUser":null}, {"uid":"uid-5_-9199836795776972030","username":"username-  
5_4090234357425186562","approvedUser":null}, {"uid":"uid-6_-5295406048781906273","username":"username-  
6_4590320736793477393","approvedUser":null}, {"uid":"uid-7_-8841575714125385615","username":"username-  
7_6846269331049068887","approvedUser":null}, {"uid":"uid-8_-5235492378307220799","username":"username-8_-  
854603671698461613","approvedUser":null}, {"uid":"uid-9_-7326522649874825763","username":"username-9_-  
770645598227583671","approvedUser":null}, {"uid":"uid-10_-1705261411298242744","username":"username-10_-  
4958056132809046934","approvedUser":null}, {"uid":"uid-11_-7999321216526831765","username":"username-  
11_6854087528646060850","approvedUser":null}, {"uid":"uid-12_-6978575702102465763","username":"username-12_-  
4670326963024762095","approvedUser":null}, {"uid":"uid-13_-8402517715372626656","username":"username-13_-  
2835814279750368691","approvedUser":null}, {"uid":"uid-14_-320254178006521591","username":"username-14_-  
4429135233197214938","approvedUser":null}, {"uid":"uid-15_-869757331514267357","username":"username-  
15_2806637997399400762","approvedUser":null}, {"uid":"uid-16_-759793596875185778","username":"username-  
16_6583323374136926506","approvedUser":null}, {"uid":"uid-17_-9030718808699054593","username":"username-17_-  
3509858659697926530","approvedUser":null}, {"uid":"uid-18_-5621927144543672518","username":"username-  
18_5757311782031866821","approvedUser":null}, {"uid":"uid-19_-3785908245864021072","username":"username-19_-  
1901098304313454336","approvedUser":null}, {"uid":"uid-20_-674940524398298009","username":"username-20_-  
2717579231691297727","approvedUser":null}, {"uid":"uid-21_-4643631466226894705","username":"username-21_-  
5076395314247796594","approvedUser":null}, {"uid":"uid-22_-3384239137568304462","username":"username-  
22_413618824165306122","approvedUser":null}, {"uid":"uid-23_-6577509033020530932","username":"username-23_-  
8618762588906228138","approvedUser":null}, {"uid":"uid-24_-1833675748964821369","username":"username-24_-  
1569605395362147276","approvedUser":null}, {"uid":"uid-25_-6445379790847017939","username":"username-25_-  
2167601777346308755","approvedUser":null}, {"uid":"uid-26_-3483057049757369704","username":"username-  
26_5116153017607294965","approvedUser":null}, {"uid":"uid-27_-8800272555618521510","username":"username-27_-  
7129618370885950644","approvedUser":null}, {"uid":"uid-28_-2144059898910177442","username":"username-28_-  
2314999609958753498","approvedUser":null}, {"uid":"uid-29_-8902380512588133718","username":"username-  
29_1200442841669720141","approvedUser":null}, {"uid":"uid-30_-2153334245696570834","username":"username-  
30_849280389414839238","approvedUser":null}, {"uid":"uid-31_-6974910607311217476","username":"username-31_-  
4324882312036298702","approvedUser":null}, {"uid":"uid-32_-3417317460305679078","username":"username-  
32_1671619512813636983","approvedUser":null}, {"uid":"uid-33_-3541803465399211295","username":"username-33_-  
1037479221311139169","approvedUser":null}, {"uid":"uid-34_-9056322543504417544","username":"username-  
34_8609794937001378303","approvedUser":null}, {"uid":"uid-35_-634788936781976654","username":"username-  
35_495229991917251483","approvedUser":null}, {"uid":"uid-36_-7120945701981424220","username":"username-  
36_5771230577124708809","approvedUser":null}, {"uid":"uid-37_-5048657096720726403","username":"username-37_-  
4530235601808763885","approvedUser":null}, {"uid":"uid-38_-2275447634146621601","username":"username-  
38_8455647715310838881","approvedUser":null}, {"uid":"uid-39_-3161980304855170610","username":"username-39_-  
6250139615484990889","approvedUser":null}, {"uid":"uid-40_-7077536951680146401","username":"username-40_-  
1435294212783032068","approvedUser":null}, {"uid":"uid-41_-8227917687462887027","username":"username-  
41_8458179260688235163","approvedUser":null}, {"uid":"uid-42_-4188872066396236605","username":"username-42_-  
226945888914340115","approvedUser":null}, {"uid":"uid-43_-9804543332400129","username":"username-  
43_5163998622589179768","approvedUser":null}, {"uid":"uid-44_-7450235681459203677","username":"username-  
44_3599563329586073137","approvedUser":null}, {"uid":"uid-45_-436210154317294925","username":"username-45_-  
2836036336133258648","approvedUser":null}, {"uid":"uid-46_-1042657016367456937","username":"username-46_-  
254450498804439068","approvedUser":null}, {"uid":"uid-47_-4394865683870473459","username":"username-
```

47_6518643561918440525","approvedUser":null,{"uid":"uid-48_3704211337738259791","username":"username-48_-
263703938164660534","approvedUser":null,{"uid":"uid-49_-3642722407600643630","username":"username-
49_5329210331475957226","approvedUser":null,{"uid":"uid-50_2607521289874358270","username":"username-
50_4831310243600398279","approvedUser":null,{"uid":"uid-51_-3033215902222982259","username":"username-
51_8177667785829660776","approvedUser":null,{"uid":"uid-52_4490230113732863341","username":"username-
52_8664940755750367847","approvedUser":null,{"uid":"uid-53_-1357194897328574708","username":"username-
53_1796522631556292051","approvedUser":null,{"uid":"uid-54_4648631375969852885","username":"username-54_-
3693288530801211049","approvedUser":null,{"uid":"uid-55_5867975711370402553","username":"username-55_-
4806914133573320293","approvedUser":null,{"uid":"uid-56_-5826015676617151630","username":"username-56_-
2841804710251197718","approvedUser":null,{"uid":"uid-57_-2707684712328954812","username":"username-57_-
5639335964680253164","approvedUser":null,{"uid":"uid-58_5420818911544979205","username":"username-58_-
2839239455911333483","approvedUser":null,{"uid":"uid-59_-7933457633045419360","username":"username-59_-
4961845167286877194","approvedUser":null,{"uid":"uid-60_-210049432158882731","username":"username-
60_2555584519510738035","approvedUser":null,{"uid":"uid-61_825343280687420675","username":"username-
61_1032865726090364125","approvedUser":null,{"uid":"uid-62_-2517045919784448343","username":"username-
62_4069026491486177217","approvedUser":null,{"uid":"uid-63_3560596193146857002","username":"username-
63_1780355201625909706","approvedUser":null,{"uid":"uid-64_3883454867283739983","username":"username-
64_9141145835174493539","approvedUser":null,{"uid":"uid-65_4769134369443641618","username":"username-
65_2440681235812875424","approvedUser":null,{"uid":"uid-66_8463069315611480089","username":"username-66_-
6427614811484097186","approvedUser":null,{"uid":"uid-67_2770995702279429504","username":"username-
67_2626821032839517096","approvedUser":null,{"uid":"uid-68_158504230936060891","username":"username-68_-
6867492938763620772","approvedUser":null,{"uid":"uid-69_2161548556092763258","username":"username-
69_3697963152167696909","approvedUser":null,{"uid":"uid-70_-3132752353413198098","username":"username-
70_5390041225037612587","approvedUser":null,{"uid":"uid-71_-139773096053166900","username":"username-
71_6898336095810644065","approvedUser":null,{"uid":"uid-72_-8893129826702858832","username":"username-
72_5208593197867416126","approvedUser":null,{"uid":"uid-73_-2193410582861330555","username":"username-
73_6649829516095507734","approvedUser":null,{"uid":"uid-74_3932786593348779778","username":"username-74_-
1144162685663299542","approvedUser":null,{"uid":"uid-75_4075460131025783927","username":"username-
75_546443300678837510","approvedUser":null,{"uid":"uid-76_8106756633595609289","username":"username-76_-
1905812317618536918","approvedUser":null,{"uid":"uid-77_-7368339022893742406","username":"username-77_-
6842921181417085972","approvedUser":null,{"uid":"uid-78_7907142585554812013","username":"username-78_-
5264712901010781284","approvedUser":null,{"uid":"uid-79_-7644969832722652887","username":"username-79_-
6252426494815933207","approvedUser":null,{"uid":"uid-80_-716602028594226340","username":"username-80_-
3225024326563208830","approvedUser":null,{"uid":"uid-81_1341062538920581859","username":"username-
81_6249454395517704960","approvedUser":null,{"uid":"uid-82_1657003163786058632","username":"username-
82_4721529499990627074","approvedUser":null,{"uid":"uid-83_6815552817810475979","username":"username-
83_1678392601275685273","approvedUser":null,{"uid":"uid-84_6884684865526225893","username":"username-84_-
3956726246769138202","approvedUser":null,{"uid":"uid-85_5151220735710017801","username":"username-85_-
3686268643511448318","approvedUser":null,{"uid":"uid-86_7466781105166760172","username":"username-86_-
6610148558347271395","approvedUser":null,{"uid":"uid-87_-713512924150662777","username":"username-
87_6102436413529950953","approvedUser":null,{"uid":"uid-88_9072189928783575323","username":"username-
88_7704004749097875256","approvedUser":null,{"uid":"uid-89_4259261291982056053","username":"username-
89_7984744861917799139","approvedUser":null,{"uid":"uid-90_5751927600908568947","username":"username-90_-
7125096552778677522","approvedUser":null,{"uid":"uid-91_4901329999018832838","username":"username-
91_5350021444116068073","approvedUser":null,{"uid":"uid-92_4678015460182477305","username":"username-
92_1852631638101605272","approvedUser":null,{"uid":"uid-93_7266815460101168305","username":"username-93_-
3252920727575338026","approvedUser":null,{"uid":"uid-94_-1168179498909799883","username":"username-94_-
2915671417734300746","approvedUser":null,{"uid":"uid-95_-4565535643820085200","username":"username-
95_6630688267574786538","approvedUser":null,{"uid":"uid-96_1752368330364835254","username":"username-
96_7757284538634937192","approvedUser":null,{"uid":"uid-97_2306827087218176388","username":"username-
97_7712861682205623626","approvedUser":null,{"uid":"uid-98_-7203187944336641909","username":"username-
98_4743974173594429685","approvedUser":null,{"uid":"uid-99_3616218166966591663","username":"username-
99_3696561251234358909","approvedUser":null,{"uid":"uid-100_-386433576600935477","username":"username-100_-

1147041754410747576","approvedUser":null},{\"uid\":\"uid-101_-3873699646883590408\",\"username\":\"username-101_-3639992631428269328\",\"approvedUser\":null},{\"uid\":\"uid-102_-1289644198300206577\",\"username\":\"username-102_-5114268581982820243\",\"approvedUser\":null},{\"uid\":\"uid-103_7465736810154885995\",\"username\":\"username-103_5933438280105022835\",\"approvedUser\":null},{\"uid\":\"uid-104_3383635178415328243\",\"username\":\"username-104_2624083101392646681\",\"approvedUser\":null},{\"uid\":\"uid-105_9075429445315866455\",\"username\":\"username-105_-3973755550656411973\",\"approvedUser\":null},{\"uid\":\"uid-106_-7021878560199246696\",\"username\":\"username-106_-8064384502445618656\",\"approvedUser\":null},{\"uid\":\"uid-107_9076902112732590798\",\"username\":\"username-107_5100669602117613022\",\"approvedUser\":null},{\"uid\":\"uid-108_3923874003954124325\",\"username\":\"username-108_-4680257690845102852\",\"approvedUser\":null},{\"uid\":\"uid-109_-7647786200639426264\",\"username\":\"username-109_-5046386373030885159\",\"approvedUser\":null},{\"uid\":\"uid-110_2688621773023352385\",\"username\":\"username-110_-5210240757947506611\",\"approvedUser\":null},{\"uid\":\"uid-111_9034455979967149440\",\"username\":\"username-111_-8086500401160708851\",\"approvedUser\":null},{\"uid\":\"uid-112_8270411290670529179\",\"username\":\"username-112_-5392277563774168236\",\"approvedUser\":null},{\"uid\":\"uid-113_7398229927122249525\",\"username\":\"username-113_-2905530470718091330\",\"approvedUser\":null},{\"uid\":\"uid-114_6726215493889154403\",\"username\":\"username-114_-8400462844191944842\",\"approvedUser\":null},{\"uid\":\"uid-115_6524285850179735147\",\"username\":\"username-115_-8300721102987274605\",\"approvedUser\":null},{\"uid\":\"uid-116_7367267664674689486\",\"username\":\"username-116_-6877981477524322113\",\"approvedUser\":null},{\"uid\":\"uid-117_-2618007122736256888\",\"username\":\"username-117_7631932871454844953\",\"approvedUser\":null},{\"uid\":\"uid-118_-6294921430552088427\",\"username\":\"username-118_-8549182393314042276\",\"approvedUser\":null},{\"uid\":\"uid-119_-6207181570634822473\",\"username\":\"username-119_736919805614044224\",\"approvedUser\":null},{\"uid\":\"uid-120_-6132937796101438346\",\"username\":\"username-120_7659399969455053600\",\"approvedUser\":null},{\"uid\":\"uid-121_-2048959491295469464\",\"username\":\"username-121_2374078318519135452\",\"approvedUser\":null},{\"uid\":\"uid-122_5875588646535251634\",\"username\":\"username-122_8734026811034895872\",\"approvedUser\":null},{\"uid\":\"uid-123_-7990869219920602162\",\"username\":\"username-123_4648559244798975525\",\"approvedUser\":null},{\"uid\":\"uid-124_-192770251771221107\",\"username\":\"username-124_6883495127477584535\",\"approvedUser\":null},{\"uid\":\"uid-125_-2714663006056718753\",\"username\":\"username-125_133060902160881463\",\"approvedUser\":null},{\"uid\":\"uid-126_2766253060598152914\",\"username\":\"username-126_-4046373442947776093\",\"approvedUser\":null},{\"uid\":\"uid-127_-4272367183521049768\",\"username\":\"username-127_6569662376508564115\",\"approvedUser\":null},{\"uid\":\"uid-128_-545905186874257331\",\"username\":\"username-128_1515372795785068934\",\"approvedUser\":null},{\"uid\":\"uid-129_6708595687218488887\",\"username\":\"username-129_-7122811480333726051\",\"approvedUser\":null},{\"uid\":\"uid-130_-8321448456492076978\",\"username\":\"username-130_2038454811704254149\",\"approvedUser\":null},{\"uid\":\"uid-131_-7091233298935642305\",\"username\":\"username-131_-2564513859762813412\",\"approvedUser\":null},{\"uid\":\"uid-132_6834179719902231596\",\"username\":\"username-132_-4694525694407838579\",\"approvedUser\":null},{\"uid\":\"uid-133_3835347599573549047\",\"username\":\"username-133_-8882276029058762564\",\"approvedUser\":null},{\"uid\":\"uid-134_-8679965084834727232\",\"username\":\"username-134_7722580925735357737\",\"approvedUser\":null},{\"uid\":\"uid-135_-7740818531972343893\",\"username\":\"username-135_8247692663347270543\",\"approvedUser\":null},{\"uid\":\"uid-136_1107535675588741350\",\"username\":\"username-136_-2244473992799509875\",\"approvedUser\":null},{\"uid\":\"uid-137_5139078247128336362\",\"username\":\"username-137_3882028351672355208\",\"approvedUser\":null},{\"uid\":\"uid-138_8215261920820172288\",\"username\":\"username-138_982657662092930766\",\"approvedUser\":null},{\"uid\":\"uid-139_-5735472520401305338\",\"username\":\"username-139_1414963137239526399\",\"approvedUser\":null},{\"uid\":\"uid-140_79102286362275954\",\"username\":\"username-140_9022842745211014312\",\"approvedUser\":null},{\"uid\":\"uid-141_-5517136444856795497\",\"username\":\"username-141_-1402727634202127568\",\"approvedUser\":null},{\"uid\":\"uid-142_7231657223947985447\",\"username\":\"username-142_-6867533128802834441\",\"approvedUser\":null},{\"uid\":\"uid-143_-5526346348210590728\",\"username\":\"username-143_8509191165339342523\",\"approvedUser\":null},{\"uid\":\"uid-144_-1986767394333991670\",\"username\":\"username-144_2731583419161476670\",\"approvedUser\":null},{\"uid\":\"uid-145_-7832056389010073399\",\"username\":\"username-145_1501092723949934117\",\"approvedUser\":null},{\"uid\":\"uid-146_-8703731771159611506\",\"username\":\"username-146_-5618309228390879453\",\"approvedUser\":null},{\"uid\":\"uid-147_1524212203697274219\",\"username\":\"username-147_4053317068912431446\",\"approvedUser\":null},{\"uid\":\"uid-148_5037040867156064551\",\"username\":\"username-148_-7632562053929247504\",\"approvedUser\":null},{\"uid\":\"uid-149_265036974732857223\",\"username\":\"username-149_5745166776627810586\",\"approvedUser\":null},{\"uid\":\"uid-150_-3603564723412453681\",\"username\":\"username-150_9171115159139310868\",\"approvedUser\":null},{\"uid\":\"uid-151_1671129302512985735\",\"username\":\"username-151_5692730563525287840\",\"approvedUser\":null},{\"uid\":\"uid-152_5754802402562932610\",\"username\":\"username-152_-5157250226374625120\",\"approvedUser\":null},{\"uid\":\"uid-153_7326741928589848604\",\"username\":\"username-

153_4605407074362724606","approvedUser":null},{ "uid":"uid-154_-2963483721498535742","username":"username-154_-5527692195574758603","approvedUser":null},{ "uid":"uid-155_9105270931237680310","username":"username-155_-8176604369005913174","approvedUser":null},{ "uid":"uid-156_-8651285404876977389","username":"username-156_-4578669497755900783","approvedUser":null},{ "uid":"uid-157_383384296797973519","username":"username-157_-7791182315452579582","approvedUser":null},{ "uid":"uid-158_-1364345736062353792","username":"username-158_-1762017442497125257","approvedUser":null},{ "uid":"uid-159_-4526818156469739103","username":"username-159_-7124788561327687110","approvedUser":null},{ "uid":"uid-160_2692084811272742677","username":"username-160_-3828900518721214908","approvedUser":null},{ "uid":"uid-161_-7923410536774244053","username":"username-161_-159669131763350168","approvedUser":null},{ "uid":"uid-162_7667451658145120610","username":"username-162_-5832652097787275057","approvedUser":null},{ "uid":"uid-163_-5317448481580866768","username":"username-163_-953942764432299139","approvedUser":null},{ "uid":"uid-164_-2514511230283374850","username":"username-164_-2687795782419063339","approvedUser":null},{ "uid":"uid-165_1609566101945291127","username":"username-165_-3251661559002830193","approvedUser":null},{ "uid":"uid-166_-770784857281740030","username":"username-166_-7393296277646227038","approvedUser":null},{ "uid":"uid-167_1160216281205668748","username":"username-167_-367007711744932684","approvedUser":null},{ "uid":"uid-168_4674171505100721802","username":"username-168_-3843213940913762918","approvedUser":null},{ "uid":"uid-169_6591751454611940158","username":"username-169_-6766966044029503618","approvedUser":null},{ "uid":"uid-170_6340120704436976952","username":"username-170_-376700204976465974","approvedUser":null},{ "uid":"uid-171_3764211758434535711","username":"username-171_-789183018750421352","approvedUser":null},{ "uid":"uid-172_998980734770557361","username":"username-172_-6955967439775444780","approvedUser":null},{ "uid":"uid-173_-1852381661411426041","username":"username-173_-7694428977109198063","approvedUser":null},{ "uid":"uid-174_-1289267198926892515","username":"username-174_-7486849982408465254","approvedUser":null},{ "uid":"uid-175_4100985653695878801","username":"username-175_-2911069316406035470","approvedUser":null},{ "uid":"uid-176_609220971571115179","username":"username-176_-2263853238115263","approvedUser":null},{ "uid":"uid-177_-430030930491457123","username":"username-177_-2005270615401590730","approvedUser":null},{ "uid":"uid-178_-1898751311956911093","username":"username-178_-4601093308891935753","approvedUser":null},{ "uid":"uid-179_1737775598420552396","username":"username-179_-6363612616248159202","approvedUser":null},{ "uid":"uid-180_4727275075335209970","username":"username-180_-5993377054529608581","approvedUser":null},{ "uid":"uid-181_5643447874315403271","username":"username-181_-3399165591174174205","approvedUser":null},{ "uid":"uid-182_-3446190671849521766","username":"username-182_-3997931068526362949","approvedUser":null},{ "uid":"uid-183_7198819021614897401","username":"username-183_-8183210355964555017","approvedUser":null},{ "uid":"uid-184_-2077895718236709825","username":"username-184_-595109391782374982","approvedUser":null},{ "uid":"uid-185_7289874106634349456","username":"username-185_-2635381837340692152","approvedUser":null},{ "uid":"uid-186_4712411434153947555","username":"username-186_-7301026133767083697","approvedUser":null},{ "uid":"uid-187_671795369356595328","username":"username-187_-5140255449050681720","approvedUser":null},{ "uid":"uid-188_-2307017146290333046","username":"username-188_-6490954318400096287","approvedUser":null},{ "uid":"uid-189_-5902476389552760896","username":"username-189_-110333606917651658","approvedUser":null},{ "uid":"uid-190_8653318980636138303","username":"username-190_-5719812867861689496","approvedUser":null},{ "uid":"uid-191_-1229739897142481654","username":"username-191_-7443103089371563393","approvedUser":null},{ "uid":"uid-192_2655813105113435150","username":"username-192_-5705046091493904884","approvedUser":null},{ "uid":"uid-193_-2071131790609413025","username":"username-193_-8777456914125575363","approvedUser":null},{ "uid":"uid-194_5642184119115240903","username":"username-194_-4594754334742521583","approvedUser":null},{ "uid":"uid-195_8927890315370497666","username":"username-195_-2283661300375565235","approvedUser":null},{ "uid":"uid-196_2830701545497951810","username":"username-196_-8894468972165202978","approvedUser":null},{ "uid":"uid-197_5605296274373514616","username":"username-197_-6796838020740713713","approvedUser":null},{ "uid":"uid-198_-1302131849769614123","username":"username-198_-5557175976871156213","approvedUser":null},{ "uid":"uid-199_6100185163775033614","username":"username-199_-7496866511495296287","approvedUser":null},{ "uid":"uid-200_-1219765826021863713","username":"username-200_-4349827285067958976","approvedUser":null},{ "uid":"uid-201_-2445862782577913005","username":"username-201_-5174125324261002595","approvedUser":null},{ "uid":"uid-202_475336074402620481","username":"username-202_-757086027275696215","approvedUser":null},{ "uid":"uid-203_-2726025835811087079","username":"username-203_-9223166571979550693","approvedUser":null},{ "uid":"uid-204_-593680929304658976","username":"username-204_-3229890174283192695","approvedUser":null},{ "uid":"uid-205_4980117128872905336","username":"username-205_-597057291787378665","approvedUser":null},{ "uid":"uid-206_-9088780382246397915","username":"username-206_-

1133280435935799243","approvedUser":null},{ "uid":"uid-207_4089099341970573132","username":"username-207_7724770157475032754","approvedUser":null},{ "uid":"uid-208_-8728105559089122436","username":"username-208_2057396582231902046","approvedUser":null},{ "uid":"uid-209_8948082615665865650","username":"username-209_-3341366578951495697","approvedUser":null},{ "uid":"uid-210_-951741255466713896","username":"username-210_6960012987279564840","approvedUser":null},{ "uid":"uid-211_1402508164157433999","username":"username-211_3179410293368899639","approvedUser":null},{ "uid":"uid-212_6052694828227077166","username":"username-212_-313575757698948770","approvedUser":null},{ "uid":"uid-213_7437605783590964233","username":"username-213_6535258512274920662","approvedUser":null},{ "uid":"uid-214_4451171529842505461","username":"username-214_-5069725695235535633","approvedUser":null},{ "uid":"uid-215_1810632641239102987","username":"username-215_-4185360074262428549","approvedUser":null},{ "uid":"uid-216_-6939159453823857492","username":"username-216_3701081967160240237","approvedUser":null},{ "uid":"uid-217_9192703281367229638","username":"username-217_-4383976924264353860","approvedUser":null},{ "uid":"uid-218_3874231106289943380","username":"username-218_3556646386852957580","approvedUser":null},{ "uid":"uid-219_1142270264310808875","username":"username-219_-4003943776060317296","approvedUser":null},{ "uid":"uid-220_1603696613673536498","username":"username-220_5560897806334905001","approvedUser":null},{ "uid":"uid-221_-5497250941265759375","username":"username-221_6636816894322016211","approvedUser":null},{ "uid":"uid-222_3156703882924820239","username":"username-222_-589681596112497855","approvedUser":null},{ "uid":"uid-223_2607958354527192981","username":"username-223_3173471206890030021","approvedUser":null},{ "uid":"uid-224_1961512648778494264","username":"username-224_4586867554955591797","approvedUser":null},{ "uid":"uid-225_-1657188583648402987","username":"username-225_-2379461549845871989","approvedUser":null},{ "uid":"uid-226_-5918105484755050626","username":"username-226_801999416455766413","approvedUser":null},{ "uid":"uid-227_-8584618043628272844","username":"username-227_153546924190025133","approvedUser":null},{ "uid":"uid-228_6544614639232044022","username":"username-228_69265821685095537","approvedUser":null},{ "uid":"uid-229_-3702015435501969926","username":"username-229_-1416127401687223402","approvedUser":null},{ "uid":"uid-230_-1006026844659174855","username":"username-230_127282458431282738","approvedUser":null},{ "uid":"uid-231_-5473541366214460879","username":"username-231_-8117368471266883560","approvedUser":null},{ "uid":"uid-232_-4742271167497307162","username":"username-232_-1567883418279748737","approvedUser":null},{ "uid":"uid-233_-2269009773062755799","username":"username-233_-660236066924897051","approvedUser":null},{ "uid":"uid-234_-6122714557725868772","username":"username-234_5788208109337895216","approvedUser":null},{ "uid":"uid-235_-4091452081109299468","username":"username-235_-6592400393546005993","approvedUser":null},{ "uid":"uid-236_2260931346670829824","username":"username-236_1226966011435906239","approvedUser":null},{ "uid":"uid-237_-1052496515575321365","username":"username-237_1101438133816993448","approvedUser":null},{ "uid":"uid-238_4829877714637874749","username":"username-238_-3474555114546346806","approvedUser":null},{ "uid":"uid-239_-1247246766473293785","username":"username-239_5421394127487471867","approvedUser":null},{ "uid":"uid-240_6505384312736025471","username":"username-240_-319481349541680749","approvedUser":null},{ "uid":"uid-241_7127665437735878920","username":"username-241_5122722836328359762","approvedUser":null},{ "uid":"uid-242_-5054875012435162121","username":"username-242_-8564440901941480064","approvedUser":null},{ "uid":"uid-243_-4211607752263989070","username":"username-243_1964909719636543573","approvedUser":null},{ "uid":"uid-244_-7543275721125147686","username":"username-244_6132384059535578523","approvedUser":null},{ "uid":"uid-245_-84206024983837301","username":"username-245_-7077733363992677487","approvedUser":null},{ "uid":"uid-246_-6669481173714938654","username":"username-246_837675343990129170","approvedUser":null},{ "uid":"uid-247_816022729994833855","username":"username-247_-3637969005469613578","approvedUser":null},{ "uid":"uid-248_-1371667730637454645","username":"username-248_-8709829647341093192","approvedUser":null},{ "uid":"uid-249_3766638900167060924","username":"username-249_-5855570754572341671","approvedUser":null},{ "uid":"uid-250_26715444917669847","username":"username-250_-3299603970334053501","approvedUser":null},{ "uid":"uid-251_4045192282349951158","username":"username-251_-2178091420004012560","approvedUser":null},{ "uid":"uid-252_-5394854839823838901","username":"username-252_6669216678834562316","approvedUser":null},{ "uid":"uid-253_-857131775580565212","username":"username-253_-7766835947343472423","approvedUser":null},{ "uid":"uid-254_335137582553936880","username":"username-254_9058046347841107218","approvedUser":null},{ "uid":"uid-255_-4864072808996207807","username":"username-255_7795635000174043795","approvedUser":null},{ "uid":"uid-256_3699420239009876115","username":"username-256_1432691725620450039","approvedUser":null},{ "uid":"uid-257_-7540543588001858676","username":"username-257_-5444032935011573118","approvedUser":null},{ "uid":"uid-258_-4333296014374711163","username":"username-258_-1711604636736916415","approvedUser":null},{ "uid":"uid-259_-459523211363456184","username":"username-

259_6871983594692062319","approvedUser":null},{ "uid":"uid-260_7547835285990816283","username":"username-260_-8842396197044327142","approvedUser":null},{ "uid":"uid-261_8986521598514281725","username":"username-261_7930841539959404806","approvedUser":null},{ "uid":"uid-262_-5452725495340024529","username":"username-262_6285917093136212177","approvedUser":null},{ "uid":"uid-263_4566539732884732149","username":"username-263_-1710975681055416745","approvedUser":null},{ "uid":"uid-264_3674475929419138973","username":"username-264_4732101142477455959","approvedUser":null},{ "uid":"uid-265_5341517271640226240","username":"username-265_-2837692902192474026","approvedUser":null},{ "uid":"uid-266_5876819189391227335","username":"username-266_5818441237730944195","approvedUser":null},{ "uid":"uid-267_6683021707550781616","username":"username-267_-5717324707761970853","approvedUser":null},{ "uid":"uid-268_1360137558507582226","username":"username-268_-3482365769153455153","approvedUser":null},{ "uid":"uid-269_-38724536746182050","username":"username-269_1037888583493802659","approvedUser":null},{ "uid":"uid-270_2381869507327405671","username":"username-270_-2781882233383801768","approvedUser":null},{ "uid":"uid-271_-8074053433721945309","username":"username-271_2125631939797517517","approvedUser":null},{ "uid":"uid-272_4024472289739304626","username":"username-272_7754958186082569441","approvedUser":null},{ "uid":"uid-273_5133065440908344224","username":"username-273_-5447604659094681507","approvedUser":null},{ "uid":"uid-274_4547644412872473447","username":"username-274_2278845930776622609","approvedUser":null},{ "uid":"uid-275_-2084777788315509449","username":"username-275_-2465174954107171268","approvedUser":null},{ "uid":"uid-276_-5954290756124283656","username":"username-276_-1192332151562747088","approvedUser":null},{ "uid":"uid-277_-5588439406805516657","username":"username-277_-5370552510515828682","approvedUser":null},{ "uid":"uid-278_-870415519219791520","username":"username-278_-2774370226146221394","approvedUser":null},{ "uid":"uid-279_4776278969558431763","username":"username-279_6864453725427534202","approvedUser":null},{ "uid":"uid-280_-4105405424558788453","username":"username-280_-4066246789615642979","approvedUser":null},{ "uid":"uid-281_-3713660482578966412","username":"username-281_7330931990805565214","approvedUser":null},{ "uid":"uid-282_1011084386694715691","username":"username-282_-2746294963923696334","approvedUser":null},{ "uid":"uid-283_-2831905537216976605","username":"username-283_-161788466152417127","approvedUser":null},{ "uid":"uid-284_2922978785768119445","username":"username-284_1848983819642198344","approvedUser":null},{ "uid":"uid-285_3747863922090656472","username":"username-285_-8994176265435434937","approvedUser":null},{ "uid":"uid-286_-63165887187921950","username":"username-286_-6761865837361155364","approvedUser":null},{ "uid":"uid-287_-9171517073280830962","username":"username-287_162284901718623209","approvedUser":null},{ "uid":"uid-288_481022239783601183","username":"username-288_-9045411465012369518","approvedUser":null},{ "uid":"uid-289_2754116099325553540","username":"username-289_-4005251339691184862","approvedUser":null},{ "uid":"uid-290_-568533263314183459","username":"username-290_-771425520663686353","approvedUser":null},{ "uid":"uid-291_6126653660078706877","username":"username-291_-4131094393201014277","approvedUser":null},{ "uid":"uid-292_9054027067080211121","username":"username-292_1118733540540971036","approvedUser":null},{ "uid":"uid-293_2936651390176822401","username":"username-293_482309585669711199","approvedUser":null},{ "uid":"uid-294_-9109984467073558603","username":"username-294_-5659735436126146582","approvedUser":null},{ "uid":"uid-295_-582852677563508902","username":"username-295_-7831567846007885920","approvedUser":null},{ "uid":"uid-296_1934612819898819589","username":"username-296_6471594440268852168","approvedUser":null},{ "uid":"uid-297_-4610005063681272808","username":"username-297_-281450720092338448","approvedUser":null},{ "uid":"uid-298_5586274144600640579","username":"username-298_8432561858131641615","approvedUser":null},{ "uid":"uid-299_2336713376667472973","username":"username-299_797298733186879465","approvedUser":null},{ "uid":"uid-300_-720502486960569324","username":"username-300_-1844733013888033353","approvedUser":null},{ "uid":"uid-301_6807671186922915026","username":"username-301_6552152094392393222","approvedUser":null},{ "uid":"uid-302_4575214009786524321","username":"username-302_-3164503970804461345","approvedUser":null},{ "uid":"uid-303_3679281145063486016","username":"username-303_4268529308230233127","approvedUser":null},{ "uid":"uid-304_-624211203150360809","username":"username-304_3605172426431214409","approvedUser":null},{ "uid":"uid-305_-435273188044868232","username":"username-305_-5235447482755242735","approvedUser":null},{ "uid":"uid-306_-1733402069539111641","username":"username-306_2546716009989989339","approvedUser":null},{ "uid":"uid-307_-70265936677963748","username":"username-307_-6616567180115002388","approvedUser":null},{ "uid":"uid-308_2849186568740911495","username":"username-308_912228245222419487","approvedUser":null},{ "uid":"uid-309_-852407642323334893","username":"username-309_5773246175395091651","approvedUser":null},{ "uid":"uid-310_-1151380655208790418","username":"username-310_2128250341766901955","approvedUser":null},{ "uid":"uid-311_-2224854068816707319","username":"username-311_-3492659519731598073","approvedUser":null},{ "uid":"uid-312_2776558675729369116","username":"username-

312_3021390292163621402","approvedUser":null,{"uid":"uid-313_4840330515200325356","username":"username-313_-
1751758019307949043","approvedUser":null,{"uid":"uid-314_3647893375255855440","username":"username-314_-
1710936977697268582","approvedUser":null,{"uid":"uid-315_-2278347905337195408","username":"username-
315_5277188404181091211","approvedUser":null,{"uid":"uid-316_2744478233058516944","username":"username-
316_838048157627951452","approvedUser":null,{"uid":"uid-317_504511970378095244","username":"username-
317_2474716171652196383","approvedUser":null,{"uid":"uid-318_9209626791978129637","username":"username-
318_4413869333255032479","approvedUser":null,{"uid":"uid-319_-8138262798606440965","username":"username-
319_1168339776777161160","approvedUser":null,{"uid":"uid-320_379788454351640028","username":"username-320_-
6118980428055003066","approvedUser":null,{"uid":"uid-321_-479979613432724547","username":"username-
321_2208482400409961579","approvedUser":null,{"uid":"uid-322_5153247979356106886","username":"username-322_-
4814523941711134635","approvedUser":null,{"uid":"uid-323_8376502108014495123","username":"username-323_-
3188178329811484593","approvedUser":null,{"uid":"uid-324_7617155617483112376","username":"username-324_-
8755561656822677641","approvedUser":null,{"uid":"uid-325_7225541681215866502","username":"username-
325_4400246178843499169","approvedUser":null,{"uid":"uid-326_-6295815733845019700","username":"username-326_-
5706683434785190533","approvedUser":null,{"uid":"uid-327_5407160006237065056","username":"username-327_-
7881906342297264655","approvedUser":null,{"uid":"uid-328_-887184206409399521","username":"username-328_-
630275078198486573","approvedUser":null,{"uid":"uid-329_-8276636305111692495","username":"username-
329_5459790965196655283","approvedUser":null,{"uid":"uid-330_-6819560322573343087","username":"username-330_-
4782808319379553261","approvedUser":null,{"uid":"uid-331_-6609236071734735924","username":"username-
331_8129568275204648896","approvedUser":null,{"uid":"uid-332_-1579943464433605112","username":"username-332_-
706229485194462088","approvedUser":null,{"uid":"uid-333_-4581713434482434489","username":"username-333_-
8381957035352452398","approvedUser":null,{"uid":"uid-334_6771056289896789229","username":"username-
334_5982723367331214431","approvedUser":null,{"uid":"uid-335_-5196397232515879041","username":"username-335_-
2799092014968575733","approvedUser":null,{"uid":"uid-336_-355173549633478995","username":"username-336_-
5561871949521936695","approvedUser":null,{"uid":"uid-337_7763295908799529105","username":"username-
337_4876183165669023599","approvedUser":null,{"uid":"uid-338_-2104539520054770887","username":"username-338_-
3112119700480769827","approvedUser":null,{"uid":"uid-339_8269711318558272354","username":"username-
339_4566875717852361573","approvedUser":null,{"uid":"uid-340_-1239381978587536506","username":"username-340_-
6745239518308511130","approvedUser":null,{"uid":"uid-341_-940411794368448743","username":"username-341_-
2951987953506084038","approvedUser":null,{"uid":"uid-342_-8168642087541637618","username":"username-342_-
401977591517270447","approvedUser":null,{"uid":"uid-343_1029672987439119519","username":"username-
343_1738509962731224734","approvedUser":null,{"uid":"uid-344_7545085320697281358","username":"username-344_-
1430232421717077199","approvedUser":null,{"uid":"uid-345_3261389632871586598","username":"username-345_-
7776546571019026652","approvedUser":null,{"uid":"uid-346_3635413103710808646","username":"username-346_-
8619613193503280197","approvedUser":null,{"uid":"uid-347_-4729931723537892193","username":"username-347_-
2628297501011554106","approvedUser":null,{"uid":"uid-348_1499520762115722492","username":"username-
348_629122289958688586","approvedUser":null,{"uid":"uid-349_-5226257005070849470","username":"username-349_-
5765130251925986720","approvedUser":null,{"uid":"uid-350_-4945274225244850951","username":"username-
350_5135298677376797276","approvedUser":null,{"uid":"uid-351_4830958770125533450","username":"username-
351_2846274377289651760","approvedUser":null,{"uid":"uid-352_-2468326279679430710","username":"username-
352_2312189505076059922","approvedUser":null,{"uid":"uid-353_-8627345676337375690","username":"username-
353_2656750641306077789","approvedUser":null,{"uid":"uid-354_-6048174407136115840","username":"username-
354_6920001696123108788","approvedUser":null,{"uid":"uid-355_7695596006128524234","username":"username-
355_6254398353941967213","approvedUser":null,{"uid":"uid-356_625478394126908208","username":"username-
356_6391918498918974763","approvedUser":null,{"uid":"uid-357_8331937228345283852","username":"username-
357_7418932817913412300","approvedUser":null,{"uid":"uid-358_-2131240883805486220","username":"username-
358_4601475333315509126","approvedUser":null,{"uid":"uid-359_-3295721782130750713","username":"username-
359_2428437344875006796","approvedUser":null,{"uid":"uid-360_6054801678289458921","username":"username-360_-
4951042783396852106","approvedUser":null,{"uid":"uid-361_-8485130698130181877","username":"username-
361_2745891781291705760","approvedUser":null,{"uid":"uid-362_7390547457748744879","username":"username-
362_7653449762288378833","approvedUser":null,{"uid":"uid-363_7972011201220372779","username":"username-
363_2122746149706359519","approvedUser":null,{"uid":"uid-364_8502335242417212140","username":"username-
364_1280224672863828498","approvedUser":null,{"uid":"uid-365_-9072330661070251843","username":"username-365_-

8040644112126452984","approvedUser":null,{"uid":"uid-366_4624401255599527753","username":"username-366_6744772918593607895","approvedUser":null,{"uid":"uid-367_-5792172483619861282","username":"username-367_3415901756532530552","approvedUser":null,{"uid":"uid-368_-7496814208827273357","username":"username-368_-7444511210871008926","approvedUser":null,{"uid":"uid-369_-7706737253517054228","username":"username-369_-3813332899631490699","approvedUser":null,{"uid":"uid-370_-8129329518189494373","username":"username-370_6755105885748711193","approvedUser":null,{"uid":"uid-371_5041114605214320906","username":"username-371_5245234110818784935","approvedUser":null,{"uid":"uid-372_-6075131809230021385","username":"username-372_145354502195574791","approvedUser":null,{"uid":"uid-373_-9039634797109765473","username":"username-373_-7604721250796675687","approvedUser":null,{"uid":"uid-374_7259008101751464511","username":"username-374_7509456043588033880","approvedUser":null,{"uid":"uid-375_-2626144210411307817","username":"username-375_23262459722592317","approvedUser":null,{"uid":"uid-376_-5234301327989867472","username":"username-376_-224882796609569823","approvedUser":null,{"uid":"uid-377_-6815858983029745888","username":"username-377_5933371565745459813","approvedUser":null,{"uid":"uid-378_1841424330156443149","username":"username-378_-2844559673944634219","approvedUser":null,{"uid":"uid-379_5341923840697934767","username":"username-379_-2422206961782660533","approvedUser":null,{"uid":"uid-380_8461081775610795545","username":"username-380_7439214234811852049","approvedUser":null,{"uid":"uid-381_3361260170558637049","username":"username-381_-2927554886447644406","approvedUser":null,{"uid":"uid-382_2201172271385172510","username":"username-382_-9174631411685021679","approvedUser":null,{"uid":"uid-383_2127703132482839866","username":"username-383_8141911255248096034","approvedUser":null,{"uid":"uid-384_-1924148209716608023","username":"username-384_-8174278125276363272","approvedUser":null,{"uid":"uid-385_5073631749222610161","username":"username-385_-2683833225318260488","approvedUser":null,{"uid":"uid-386_-8876087920228497973","username":"username-386_4262970701861375309","approvedUser":null,{"uid":"uid-387_7963111340983690594","username":"username-387_8965507896135335763","approvedUser":null,{"uid":"uid-388_-6204465525784174823","username":"username-388_2622026347286104888","approvedUser":null,{"uid":"uid-389_3747009788016296467","username":"username-389_-3379245064649398304","approvedUser":null,{"uid":"uid-390_-5046301333824440147","username":"username-390_4280229403622128649","approvedUser":null,{"uid":"uid-391_7149597604763718735","username":"username-391_-2481358707959064144","approvedUser":null,{"uid":"uid-392_5977829508029583716","username":"username-392_-3752720214038377550","approvedUser":null,{"uid":"uid-393_428168724908382118","username":"username-393_-8825681675622983243","approvedUser":null,{"uid":"uid-394_3871052733872061086","username":"username-394_-6798068376970189240","approvedUser":null,{"uid":"uid-395_-2150112737956938133","username":"username-395_-938949638786446520","approvedUser":null,{"uid":"uid-396_-1279912894731885375","username":"username-396_7313814691595425925","approvedUser":null,{"uid":"uid-397_-7843039475830176204","username":"username-397_-2748294731736770394","approvedUser":null,{"uid":"uid-398_1578103500472810422","username":"username-398_1685611259301281865","approvedUser":null,{"uid":"uid-399_4955839094778518622","username":"username-399_-6802460180925666450","approvedUser":null,{"uid":"uid-400_9042797244065764203","username":"username-400_9012547203705589013","approvedUser":null,{"uid":"uid-401_6787531294631793287","username":"username-401_7293799262420874532","approvedUser":null,{"uid":"uid-402_-1894148203256961969","username":"username-402_8723915461967106190","approvedUser":null,{"uid":"uid-403_1628931630596629266","username":"username-403_5351806108197324722","approvedUser":null,{"uid":"uid-404_3338706133248907224","username":"username-404_-9179293699442057472","approvedUser":null,{"uid":"uid-405_257220329741043110","username":"username-405_-1900908322456928798","approvedUser":null,{"uid":"uid-406_963776619134772480","username":"username-406_-3284446698400071720","approvedUser":null,{"uid":"uid-407_3316230321756148655","username":"username-407_4960403954396672997","approvedUser":null,{"uid":"uid-408_8132808304809147588","username":"username-408_-6668083097817895383","approvedUser":null,{"uid":"uid-409_2315672959055789276","username":"username-409_-5053415813649477681","approvedUser":null,{"uid":"uid-410_-6409097427765213532","username":"username-410_3614534096192121128","approvedUser":null,{"uid":"uid-411_-6086181164396786785","username":"username-411_-5329997345055909352","approvedUser":null,{"uid":"uid-412_5345736456845815079","username":"username-412_-7805696767970945872","approvedUser":null,{"uid":"uid-413_-2552075244844762704","username":"username-413_-2833425497228663268","approvedUser":null,{"uid":"uid-414_-5320868321699382700","username":"username-414_1885600848542624191","approvedUser":null,{"uid":"uid-415_-6002593439099120501","username":"username-415_-3096763635956168036","approvedUser":null,{"uid":"uid-416_9201961026377548857","username":"username-416_7369311175596750583","approvedUser":null,{"uid":"uid-417_-8432407322982714899","username":"username-417_-437321027636647365","approvedUser":null,{"uid":"uid-418_-149831917645026969","username":"username-418_-

7992227819847143138","approvedUser":null},{ "uid":"uid-419_-4644300149602945957","username":"username-419_-
741723952209884157","approvedUser":null},{ "uid":"uid-420_8666134319203523956","username":"username-
420_6878894652414956446","approvedUser":null},{ "uid":"uid-421_244975841387814811","username":"username-421_-
1788006158905097251","approvedUser":null},{ "uid":"uid-422_-839324466037815944","username":"username-
422_1903317550353355014","approvedUser":null},{ "uid":"uid-423_7182644702147657016","username":"username-423_-
1394086780274676964","approvedUser":null},{ "uid":"uid-424_-1141821591293729648","username":"username-
424_8923880723578557319","approvedUser":null},{ "uid":"uid-425_-6005540286265306878","username":"username-
425_2909191985974749427","approvedUser":null},{ "uid":"uid-426_-8179167378943158177","username":"username-
426_1875825809061383543","approvedUser":null},{ "uid":"uid-427_2727732806118841740","username":"username-427_-
6788273759238124806","approvedUser":null},{ "uid":"uid-428_-2531991403451914872","username":"username-428_-
2341981233509082670","approvedUser":null},{ "uid":"uid-429_6778000304935844013","username":"username-429_-
9101682870341571155","approvedUser":null},{ "uid":"uid-430_-2089887040365243339","username":"username-
430_7507130058450578612","approvedUser":null},{ "uid":"uid-431_2537106589518082047","username":"username-431_-
1816484999904529171","approvedUser":null},{ "uid":"uid-432_2142702425893149843","username":"username-432_-
8063075864074021920","approvedUser":null},{ "uid":"uid-433_-8696147591197377181","username":"username-433_-
597859099644095363","approvedUser":null},{ "uid":"uid-434_-2301172753695546541","username":"username-
434_1938495363034548028","approvedUser":null},{ "uid":"uid-435_74306776228072315","username":"username-435_-
1864820905982056314","approvedUser":null},{ "uid":"uid-436_-2995437071430613461","username":"username-
436_2230258180164782419","approvedUser":null},{ "uid":"uid-437_5435571962601277999","username":"username-
437_5668311769141059091","approvedUser":null},{ "uid":"uid-438_-7949932777673410477","username":"username-438_-
3897307461413677103","approvedUser":null},{ "uid":"uid-439_-6476485694710688983","username":"username-439_-
4622426381261853802","approvedUser":null},{ "uid":"uid-440_-8056997704586678519","username":"username-
440_2878798044068850699","approvedUser":null},{ "uid":"uid-441_8919179343023677016","username":"username-
441_4179717972716034401","approvedUser":null},{ "uid":"uid-442_-5393248046594892275","username":"username-442_-
3317054877045355659","approvedUser":null},{ "uid":"uid-443_4808198748884125297","username":"username-
443_2963913369563699758","approvedUser":null},{ "uid":"uid-444_9147881776329966629","username":"username-444_-
3850496356055865303","approvedUser":null},{ "uid":"uid-445_-6905905821095371415","username":"username-445_-
5688102256073362536","approvedUser":null},{ "uid":"uid-446_-7558908141936261221","username":"username-
446_6674559897201856871","approvedUser":null},{ "uid":"uid-447_8363259751943491836","username":"username-
447_2549679132136945311","approvedUser":null},{ "uid":"uid-448_-8696138026806369606","username":"username-448_-
9101042083966258161","approvedUser":null},{ "uid":"uid-449_8720261770926497578","username":"username-
449_2160385826321188904","approvedUser":null},{ "uid":"uid-450_8689192150871531604","username":"username-
450_3569296771102550825","approvedUser":null},{ "uid":"uid-451_-8041660983216530764","username":"username-
451_2319129908234189179","approvedUser":null},{ "uid":"uid-452_-2141370795422324673","username":"username-452_-
31596082190655395","approvedUser":null},{ "uid":"uid-453_-6293029732937941085","username":"username-
453_6618154167893127363","approvedUser":null},{ "uid":"uid-454_-523166080709092397","username":"username-454_-
7631827608204870533","approvedUser":null},{ "uid":"uid-455_7546785008862336962","username":"username-
455_2800817929194119570","approvedUser":null},{ "uid":"uid-456_8925137417883091566","username":"username-456_-
975408656315799563","approvedUser":null},{ "uid":"uid-457_8116818936632782623","username":"username-457_-
5116871701891148454","approvedUser":null},{ "uid":"uid-458_4583687739546732016","username":"username-458_-
4567595125819401376","approvedUser":null},{ "uid":"uid-459_939462301592328334","username":"username-459_-
3399161653618424844","approvedUser":null},{ "uid":"uid-460_7602008606338531135","username":"username-460_-
2913483869455253513","approvedUser":null},{ "uid":"uid-461_-5756286443069930970","username":"username-
461_8252565089740028957","approvedUser":null},{ "uid":"uid-462_-1772535279039229244","username":"username-462_-
6632154339159580648","approvedUser":null},{ "uid":"uid-463_6924278512532871618","username":"username-
463_3567023056051624948","approvedUser":null},{ "uid":"uid-464_1536996482021547245","username":"username-
464_4654941223483975620","approvedUser":null},{ "uid":"uid-465_-8043629508493311615","username":"username-
465_6779275912893323646","approvedUser":null},{ "uid":"uid-466_-2234159401967063383","username":"username-
466_1770627266556867875","approvedUser":null},{ "uid":"uid-467_-8654841155706874810","username":"username-467_-
64978351117149405","approvedUser":null},{ "uid":"uid-468_-341758174858503697","username":"username-
468_3567921930582653182","approvedUser":null},{ "uid":"uid-469_-8330190171293861320","username":"username-469_-
5281080776064999378","approvedUser":null},{ "uid":"uid-470_-3398505789091508681","username":"username-470_-
6341904311737276605","approvedUser":null},{ "uid":"uid-471_-8675054643297710781","username":"username-471_-

7999476147128125104","approvedUser":null},{ "uid":"uid-472_1503221869111526655","username":"username-472_8870501103937058693","approvedUser":null},{ "uid":"uid-473_-4575190404500615786","username":"username-473_4759045634531774749","approvedUser":null},{ "uid":"uid-474_-6836437228442337721","username":"username-474_9210080812642792826","approvedUser":null},{ "uid":"uid-475_-8056724370565978589","username":"username-475_-8588525939102055283","approvedUser":null},{ "uid":"uid-476_5846749960913931541","username":"username-476_4003776373908988283","approvedUser":null},{ "uid":"uid-477_-5829992396034260701","username":"username-477_4800865977427168125","approvedUser":null},{ "uid":"uid-478_5068876634270823410","username":"username-478_5847248995010257159","approvedUser":null},{ "uid":"uid-479_-3136894782914728817","username":"username-479_4273951778155327788","approvedUser":null},{ "uid":"uid-480_-5399164375170068116","username":"username-480_-3601170749064233152","approvedUser":null},{ "uid":"uid-481_-1771862780244652376","username":"username-481_-1409415846034498370","approvedUser":null},{ "uid":"uid-482_-928489483877835630","username":"username-482_2409148008667544940","approvedUser":null},{ "uid":"uid-483_-1305282450477569425","username":"username-483_3303125379030051260","approvedUser":null},{ "uid":"uid-484_144853552034132347","username":"username-484_2450834430292849525","approvedUser":null},{ "uid":"uid-485_4285152252128703533","username":"username-485_5813011719544852626","approvedUser":null},{ "uid":"uid-486_4062267545227177407","username":"username-486_5895084334078378090","approvedUser":null},{ "uid":"uid-487_2219608508384671213","username":"username-487_4436325159290150672","approvedUser":null},{ "uid":"uid-488_-78725136963134212","username":"username-488_-3178717679825990916","approvedUser":null},{ "uid":"uid-489_8213611180853560317","username":"username-489_-746506741136998166","approvedUser":null},{ "uid":"uid-490_-3538132455125994158","username":"username-490_7052653748620988930","approvedUser":null},{ "uid":"uid-491_-46097904229760655","username":"username-491_5002086081624709176","approvedUser":null},{ "uid":"uid-492_-4513545871658638360","username":"username-492_-5244857547998688011","approvedUser":null},{ "uid":"uid-493_-6814063906908867424","username":"username-493_-8073678450817243657","approvedUser":null},{ "uid":"uid-494_513524711051618661","username":"username-494_3669039244083862644","approvedUser":null},{ "uid":"uid-495_7056824248164150181","username":"username-495_6319298693850219141","approvedUser":null},{ "uid":"uid-496_-7818390639887694217","username":"username-496_-2010376304766121278","approvedUser":null},{ "uid":"uid-497_-6834455467680517921","username":"username-497_-7935837979082283479","approvedUser":null},{ "uid":"uid-498_2042058186291689552","username":"username-498_-3414868206114628541","approvedUser":null},{ "uid":"uid-499_8947031008031821681","username":"username-499_-2275844922395721335","approvedUser":null},{ "uid":"uid-500_6029187652669965603","username":"username-500_611475031119572995","approvedUser":null},{ "uid":"uid-501_4434144952207714243","username":"username-501_-8803450478687853007","approvedUser":null},{ "uid":"uid-502_-1718615918466475860","username":"username-502_297936300954349615","approvedUser":null},{ "uid":"uid-503_-6804077595077291009","username":"username-503_4995191838470487827","approvedUser":null},{ "uid":"uid-504_7710566306085400741","username":"username-504_-2422960449908239719","approvedUser":null},{ "uid":"uid-505_2040518674887023765","username":"username-505_7583254074511366402","approvedUser":null},{ "uid":"uid-506_-5524910568119772846","username":"username-506_3947946290855094584","approvedUser":null},{ "uid":"uid-507_2071086527924436151","username":"username-507_-2836231347075843275","approvedUser":null},{ "uid":"uid-508_-5206113564753643018","username":"username-508_-1438807232878378330","approvedUser":null},{ "uid":"uid-509_9073147693085865693","username":"username-509_8965813239367421899","approvedUser":null},{ "uid":"uid-510_2374147416695615835","username":"username-510_7567297886085290656","approvedUser":null},{ "uid":"uid-511_-859248307457296061","username":"username-511_3711329350204349505","approvedUser":null},{ "uid":"uid-512_610876568428757583","username":"username-512_-927250058449446262","approvedUser":null},{ "uid":"uid-513_5250037333834240614","username":"username-513_4380816859277216253","approvedUser":null},{ "uid":"uid-514_8133799121797500961","username":"username-514_4821782559272954344","approvedUser":null},{ "uid":"uid-515_-6402146821352114805","username":"username-515_5556483159639703239","approvedUser":null},{ "uid":"uid-516_367162181082524456","username":"username-516_-5487178260640706842","approvedUser":null},{ "uid":"uid-517_886346705117138744","username":"username-517_9094405358498022730","approvedUser":null},{ "uid":"uid-518_-7596222803285660356","username":"username-518_-8994331178493993263","approvedUser":null},{ "uid":"uid-519_3577035571281422777","username":"username-519_-2049886669095418472","approvedUser":null},{ "uid":"uid-520_-6723107384967797189","username":"username-520_-1662924920635326820","approvedUser":null},{ "uid":"uid-521_-1830428445797801572","username":"username-521_-2842501413344981597","approvedUser":null},{ "uid":"uid-522_-2215987288426653562","username":"username-522_5872992971504137052","approvedUser":null},{ "uid":"uid-523_-2248079960082732491","username":"username-523_819107461511440837","approvedUser":null},{ "uid":"uid-524_9168624517648500097","username":"username-524_-

6798946204687433012","approvedUser":null},{ "uid":"uid-525_9005567997846675995","username":"username-525_-
2413553748773319367","approvedUser":null},{ "uid":"uid-526_3621979497860729366","username":"username-
526_4139736020300243914","approvedUser":null},{ "uid":"uid-527_-3994860078745126999","username":"username-527_-
374824788452439393","approvedUser":null},{ "uid":"uid-528_-4464728096707998925","username":"username-
528_6833816921824476888","approvedUser":null},{ "uid":"uid-529_8162486235758566782","username":"username-529_-
3276612864013853572","approvedUser":null},{ "uid":"uid-530_-71185350223383993","username":"username-530_-
3260967546105430372","approvedUser":null},{ "uid":"uid-531_-1933499949191999747","username":"username-
531_7331089400574410167","approvedUser":null},{ "uid":"uid-532_6781220689901091273","username":"username-532_-
7511995799714811808","approvedUser":null},{ "uid":"uid-533_199274828123612366","username":"username-533_-
3528179125962485218","approvedUser":null},{ "uid":"uid-534_-7279746556387128911","username":"username-
534_6632783097351271571","approvedUser":null},{ "uid":"uid-535_3922111580861020507","username":"username-535_-
43484864193257761","approvedUser":null},{ "uid":"uid-536_-3219996861645790211","username":"username-536_-
8363880259905388730","approvedUser":null},{ "uid":"uid-537_-4104523719627170476","username":"username-537_-
7581228123319758065","approvedUser":null},{ "uid":"uid-538_-8705247953768275666","username":"username-538_-
3662715397550020433","approvedUser":null},{ "uid":"uid-539_7027414597044165676","username":"username-539_-
67388613770502105","approvedUser":null},{ "uid":"uid-540_-3792860392778711864","username":"username-540_-
8716645489240896087","approvedUser":null},{ "uid":"uid-541_-6341373826428104551","username":"username-541_-
9045251676646727039","approvedUser":null},{ "uid":"uid-542_-4320630714130075179","username":"username-
542_2719150451178799154","approvedUser":null},{ "uid":"uid-543_-6457360716841973480","username":"username-543_-
9071534634131105313","approvedUser":null},{ "uid":"uid-544_3431659060167092067","username":"username-544_-
316401458146375699","approvedUser":null},{ "uid":"uid-545_-7789722406976750490","username":"username-
545_4646944460445708773","approvedUser":null},{ "uid":"uid-546_7085664842418777345","username":"username-546_-
193534367557078370","approvedUser":null},{ "uid":"uid-547_3867686644592004616","username":"username-547_-
3394595625458214246","approvedUser":null},{ "uid":"uid-548_-7727347753097132218","username":"username-
548_1860008718304801968","approvedUser":null},{ "uid":"uid-549_-8323617603240795452","username":"username-
549_6626308580731367659","approvedUser":null},{ "uid":"uid-550_-3404163199140721589","username":"username-550_-
7350761173637736014","approvedUser":null},{ "uid":"uid-551_3770446524170503462","username":"username-551_-
786382596187359448","approvedUser":null},{ "uid":"uid-552_1817039535781056555","username":"username-552_-
4101238364297444139","approvedUser":null},{ "uid":"uid-553_159679162068720124","username":"username-553_-
4618518218978632315","approvedUser":null},{ "uid":"uid-554_-6728063623107386998","username":"username-
554_1324674847156126064","approvedUser":null},{ "uid":"uid-555_-456212724948933414","username":"username-555_-
7138728991284421989","approvedUser":null},{ "uid":"uid-556_1405260500439094795","username":"username-
556_2001658352377029501","approvedUser":null},{ "uid":"uid-557_-6303552751690638185","username":"username-557_-
6077094771771485212","approvedUser":null},{ "uid":"uid-558_-3675918031927662887","username":"username-
558_4622845448961308697","approvedUser":null},{ "uid":"uid-559_6134654329668669390","username":"username-559_-
5787867636225501630","approvedUser":null},{ "uid":"uid-560_3266529310696889631","username":"username-560_-
3764834425558481401","approvedUser":null},{ "uid":"uid-561_3472601490070411614","username":"username-561_-
3640479489171759496","approvedUser":null},{ "uid":"uid-562_-7565653247981400432","username":"username-562_-
8653000803262685650","approvedUser":null},{ "uid":"uid-563_-1690765154644974357","username":"username-
563_2535350824873472431","approvedUser":null},{ "uid":"uid-564_-3993473588584535584","username":"username-
564_2226325287333788695","approvedUser":null},{ "uid":"uid-565_-7723046250131389605","username":"username-
565_8669382055024838131","approvedUser":null},{ "uid":"uid-566_-4990201216711552184","username":"username-566_-
492672422248061328","approvedUser":null},{ "uid":"uid-567_-2023424153498661031","username":"username-567_-
6975544173162480972","approvedUser":null},{ "uid":"uid-568_-3814898025427906298","username":"username-
568_6818584521763667851","approvedUser":null},{ "uid":"uid-569_-1562106881367842347","username":"username-
569_3015793436574736710","approvedUser":null},{ "uid":"uid-570_8303114413253396443","username":"username-570_-
7173723748279093793","approvedUser":null},{ "uid":"uid-571_-5763872789212834175","username":"username-571_-
4372673155564268191","approvedUser":null},{ "uid":"uid-572_3324418393642239883","username":"username-572_-
2498418751442647930","approvedUser":null},{ "uid":"uid-573_-2916245860947845034","username":"username-573_-
3046109180883458655","approvedUser":null},{ "uid":"uid-574_8418312622522017632","username":"username-574_-
4539515435139722306","approvedUser":null},{ "uid":"uid-575_6068921646806740493","username":"username-
575_2645927777813482503","approvedUser":null},{ "uid":"uid-576_-1212217284480355672","username":"username-576_-
234504961102116504","approvedUser":null},{ "uid":"uid-577_3089984996413709991","username":"username-577_-

6131062996765788554","approvedUser":null,{"uid":"uid-578_937355675987731762","username":"username-578_-
1933565049651848820","approvedUser":null,{"uid":"uid-579_-6538764963267213850","username":"username-
579_6378606521651830583","approvedUser":null,{"uid":"uid-580_4453897133862287317","username":"username-580_-
2857743077375908141","approvedUser":null,{"uid":"uid-581_3903965615544942801","username":"username-
581_5214621394760070517","approvedUser":null,{"uid":"uid-582_-1845881216992559894","username":"username-582_-
1712216000934726901","approvedUser":null,{"uid":"uid-583_-8555956495764563404","username":"username-583_-
6925899604278536766","approvedUser":null,{"uid":"uid-584_362013146335825101","username":"username-
584_8689224500486578087","approvedUser":null,{"uid":"uid-585_-2420457401615220963","username":"username-585_-
7650278267445570928","approvedUser":null,{"uid":"uid-586_3321590233963381343","username":"username-586_-
100772983957455299","approvedUser":null,{"uid":"uid-587_-2406948082437225632","username":"username-
587_779235450096291001","approvedUser":null,{"uid":"uid-588_3177250635697817479","username":"username-588_-
5459939999410114298","approvedUser":null,{"uid":"uid-589_-8310079902356900331","username":"username-589_-
5341716899245351944","approvedUser":null,{"uid":"uid-590_1696129290032787884","username":"username-590_-
2606139270178301087","approvedUser":null,{"uid":"uid-591_-3304017807356869099","username":"username-
591_2785856041088272238","approvedUser":null,{"uid":"uid-592_3194474992334925373","username":"username-
592_2628997881298625104","approvedUser":null,{"uid":"uid-593_7866672512983201408","username":"username-593_-
4172863117569815010","approvedUser":null,{"uid":"uid-594_-332216811694227671","username":"username-594_-
2719487612312699655","approvedUser":null,{"uid":"uid-595_-5603338489455451335","username":"username-595_-
8263238229305491600","approvedUser":null,{"uid":"uid-596_-2833610618855307417","username":"username-596_-
3371061308514757338","approvedUser":null,{"uid":"uid-597_-6249621669723794072","username":"username-
597_7103472106753188786","approvedUser":null,{"uid":"uid-598_-8143345963658385552","username":"username-
598_7276033035255494598","approvedUser":null,{"uid":"uid-599_-8422631006764849161","username":"username-599_-
1573927952863431534","approvedUser":null,{"uid":"uid-600_8365047994932259264","username":"username-
600_8793856765453686122","approvedUser":null,{"uid":"uid-601_-8528081722568066026","username":"username-
601_2416630124627076517","approvedUser":null,{"uid":"uid-602_-1078590685685711016","username":"username-
602_2976214972985184968","approvedUser":null,{"uid":"uid-603_1873281828252057016","username":"username-
603_5302053745915329764","approvedUser":null,{"uid":"uid-604_8344426122328853567","username":"username-604_-
6817645796714256068","approvedUser":null,{"uid":"uid-605_-5942036841694578669","username":"username-
605_8942766310954826538","approvedUser":null,{"uid":"uid-606_-8603677496998542340","username":"username-606_-
3118856177539082430","approvedUser":null,{"uid":"uid-607_-2036380705709491676","username":"username-607_-
4006705763353389897","approvedUser":null,{"uid":"uid-608_-8475237722473589218","username":"username-608_-
4717267420242337539","approvedUser":null,{"uid":"uid-609_-4322980794483029207","username":"username-609_-
2780078116544202320","approvedUser":null,{"uid":"uid-610_-899737660934739276","username":"username-
610_7221088396745433268","approvedUser":null,{"uid":"uid-611_8937162095155900197","username":"username-611_-
590837191103844708","approvedUser":null,{"uid":"uid-612_-2837528231359631659","username":"username-
612_4293042784981624655","approvedUser":null,{"uid":"uid-613_8384321615034596191","username":"username-613_-
5704014051247098648","approvedUser":null,{"uid":"uid-614_-9150283952450883302","username":"username-
614_5341878625524772062","approvedUser":null,{"uid":"uid-615_5345700383849511651","username":"username-615_-
8422810617770048668","approvedUser":null,{"uid":"uid-616_7008325689995120159","username":"username-
616_2619392075319349522","approvedUser":null,{"uid":"uid-617_7102994021266776831","username":"username-617_-
546513433146683707","approvedUser":null,{"uid":"uid-618_-736666068479251194","username":"username-618_-
1192615805402567742","approvedUser":null,{"uid":"uid-619_6380644940196386760","username":"username-
619_2607021257220792865","approvedUser":null,{"uid":"uid-620_2056412576105499635","username":"username-620_-
6102289593075274392","approvedUser":null,{"uid":"uid-621_8998250266886652051","username":"username-
621_6089854592334022518","approvedUser":null,{"uid":"uid-622_-7620267725184361327","username":"username-
622_6160022781056421794","approvedUser":null,{"uid":"uid-623_-330505470138930052","username":"username-
623_6019053888499712035","approvedUser":null,{"uid":"uid-624_5229484441127645323","username":"username-624_-
1629951088371083855","approvedUser":null,{"uid":"uid-625_-48988818335496478","username":"username-625_-
4778367930912912020","approvedUser":null,{"uid":"uid-626_-5453841953772030444","username":"username-626_-
2513437708520383235","approvedUser":null,{"uid":"uid-627_6670301509199567849","username":"username-
627_2391000602234833146","approvedUser":null,{"uid":"uid-628_-3201793295674080160","username":"username-
628_9058169604835900814","approvedUser":null,{"uid":"uid-629_-2701938834255696650","username":"username-629_-
7946287779727356583","approvedUser":null,{"uid":"uid-630_4416612603359197224","username":"username-

630_8675375779623860885","approvedUser":null,{"uid":"uid-631_4064665249477317943","username":"username-631_-5535666382693388679","approvedUser":null,{"uid":"uid-632_-228112337095139286","username":"username-632_-977408510387305759","approvedUser":null,{"uid":"uid-633_7661977332033650201","username":"username-633_4965222559488606766","approvedUser":null,{"uid":"uid-634_379480884340128167","username":"username-634_-3773048664121278101","approvedUser":null,{"uid":"uid-635_6457279319304823760","username":"username-635_-1813927610870237008","approvedUser":null,{"uid":"uid-636_-8526093799094895388","username":"username-636_819968242174148043","approvedUser":null,{"uid":"uid-637_-6206112561806399022","username":"username-637_2916742386281301276","approvedUser":null,{"uid":"uid-638_-3694994821822727767","username":"username-638_215325214782934207","approvedUser":null,{"uid":"uid-639_-6887554118186041508","username":"username-639_-1034591600701172240","approvedUser":null,{"uid":"uid-640_3103017440894813125","username":"username-640_7182197341596897451","approvedUser":null,{"uid":"uid-641_5857286898260956140","username":"username-641_-338488660443685485","approvedUser":null,{"uid":"uid-642_4434163995689168776","username":"username-642_7018171648737298964","approvedUser":null,{"uid":"uid-643_-6741044754225849403","username":"username-643_6844014449009928907","approvedUser":null,{"uid":"uid-644_-1069849048291779001","username":"username-644_-45331069341779932","approvedUser":null,{"uid":"uid-645_-8443211331973110674","username":"username-645_4160322500427305284","approvedUser":null,{"uid":"uid-646_6649875845666364123","username":"username-646_-3597130638019016406","approvedUser":null,{"uid":"uid-647_3470372026823899443","username":"username-647_-44308914708798960","approvedUser":null,{"uid":"uid-648_-1611026480861329886","username":"username-648_-5332365144331740394","approvedUser":null,{"uid":"uid-649_3165730574558058402","username":"username-649_7039849237281272866","approvedUser":null,{"uid":"uid-650_3089645227898524541","username":"username-650_-1915236041302995967","approvedUser":null,{"uid":"uid-651_5535265705955210256","username":"username-651_-7015453370855670574","approvedUser":null,{"uid":"uid-652_5039715061730317649","username":"username-652_-7273635552845522269","approvedUser":null,{"uid":"uid-653_-6961507382310744558","username":"username-653_-6921599843424733092","approvedUser":null,{"uid":"uid-654_8756269664055941510","username":"username-654_5932275637006948059","approvedUser":null,{"uid":"uid-655_7355471952753480964","username":"username-655_-7672752794046258592","approvedUser":null,{"uid":"uid-656_-6473027224245889525","username":"username-656_-3928143328128081365","approvedUser":null,{"uid":"uid-657_-1328202860875735931","username":"username-657_-7238473233585947037","approvedUser":null,{"uid":"uid-658_2522120981146756431","username":"username-658_8433557106977898554","approvedUser":null,{"uid":"uid-659_-2871330961123254023","username":"username-659_-8341815000675225523","approvedUser":null,{"uid":"uid-660_7349710046962815690","username":"username-660_3967891214745843536","approvedUser":null,{"uid":"uid-661_-8088807335020127226","username":"username-661_6094153604924908778","approvedUser":null,{"uid":"uid-662_3554896727978456946","username":"username-662_-7177276710696779555","approvedUser":null,{"uid":"uid-663_831144827457431831","username":"username-663_187402073699076713","approvedUser":null,{"uid":"uid-664_598238866347288134","username":"username-664_8415607579136100850","approvedUser":null,{"uid":"uid-665_-323818415738390052","username":"username-665_5402468087727823072","approvedUser":null,{"uid":"uid-666_-6364178158055921465","username":"username-666_623628888787567558","approvedUser":null,{"uid":"uid-667_-5093542011317675320","username":"username-667_8279072865128082117","approvedUser":null,{"uid":"uid-668_2208608201379908474","username":"username-668_-9138256768985758670","approvedUser":null,{"uid":"uid-669_1384493418392831890","username":"username-669_7109564601224832374","approvedUser":null,{"uid":"uid-670_-1251222929749738905","username":"username-670_-7052745114196511178","approvedUser":null,{"uid":"uid-671_2526449593872175029","username":"username-671_2624219308546306483","approvedUser":null,{"uid":"uid-672_5304886299659251101","username":"username-672_1265323972364900978","approvedUser":null,{"uid":"uid-673_-1688593039881336081","username":"username-673_4166338401034634271","approvedUser":null,{"uid":"uid-674_2294473969112049068","username":"username-674_-1667057857975412819","approvedUser":null,{"uid":"uid-675_5426472201222495685","username":"username-675_199065866566940098","approvedUser":null,{"uid":"uid-676_3382434644684027945","username":"username-676_-629083692597404846","approvedUser":null,{"uid":"uid-677_9095668008363052542","username":"username-677_-1847331229196833655","approvedUser":null,{"uid":"uid-678_-5915218976661263315","username":"username-678_6678783170887518702","approvedUser":null,{"uid":"uid-679_8904664615939303204","username":"username-679_1065151604103959493","approvedUser":null,{"uid":"uid-680_-9045596772947433770","username":"username-680_-8801535110563774540","approvedUser":null,{"uid":"uid-681_8469652507116809990","username":"username-681_5740735377506428072","approvedUser":null,{"uid":"uid-682_2715389395044001733","username":"username-682_-5749201512216436487","approvedUser":null,{"uid":"uid-683_-6555768077209212359","username":"username-

683_4486675327771066059","approvedUser":null},{ "uid":"uid-684_9149779566646056351","username":"username-684_2247244175398943415","approvedUser":null},{ "uid":"uid-685_-3076192424973340526","username":"username-685_-7389025818924658517","approvedUser":null},{ "uid":"uid-686_420181183292234381","username":"username-686_8917918552262947598","approvedUser":null},{ "uid":"uid-687_7538210519709110701","username":"username-687_-2355437142600183769","approvedUser":null},{ "uid":"uid-688_7768360825231511739","username":"username-688_5372205736738113983","approvedUser":null},{ "uid":"uid-689_6245463302351456618","username":"username-689_6117514344052359940","approvedUser":null},{ "uid":"uid-690_-1602537479269155108","username":"username-690_-16721613604738897","approvedUser":null},{ "uid":"uid-691_-3174902722901425046","username":"username-691_-3140871704129480855","approvedUser":null},{ "uid":"uid-692_885979126434800804","username":"username-692_-310335044660280214","approvedUser":null},{ "uid":"uid-693_-1942253078611610474","username":"username-693_7848372699668057187","approvedUser":null},{ "uid":"uid-694_-8872876401352435891","username":"username-694_-6507730987922387810","approvedUser":null},{ "uid":"uid-695_2273076068972623257","username":"username-695_4483958148112132706","approvedUser":null},{ "uid":"uid-696_1590816247809388051","username":"username-696_-1240900869121921893","approvedUser":null},{ "uid":"uid-697_4477642450859229537","username":"username-697_195268610601323450","approvedUser":null},{ "uid":"uid-698_3105283591882325688","username":"username-698_-4588292634469868582","approvedUser":null},{ "uid":"uid-699_-296800521553496926","username":"username-699_7499750307526358308","approvedUser":null},{ "uid":"uid-700_806580990798522881","username":"username-700_4494164294250282611","approvedUser":null},{ "uid":"uid-701_-2373046534396058797","username":"username-701_-5804804178501157318","approvedUser":null},{ "uid":"uid-702_-2027425034597107658","username":"username-702_5810352123697059237","approvedUser":null},{ "uid":"uid-703_4165886871820893486","username":"username-703_874573259131603454","approvedUser":null},{ "uid":"uid-704_1515591427466741158","username":"username-704_-7472508257182154670","approvedUser":null},{ "uid":"uid-705_-6087463279249604044","username":"username-705_-2293127458670407853","approvedUser":null},{ "uid":"uid-706_4919717640308080417","username":"username-706_-3172899215034414208","approvedUser":null},{ "uid":"uid-707_-2614370160031529177","username":"username-707_4921952944837915010","approvedUser":null},{ "uid":"uid-708_-4820718131855266180","username":"username-708_-5294256343112985895","approvedUser":null},{ "uid":"uid-709_-286599146838619633","username":"username-709_4682692174576363703","approvedUser":null},{ "uid":"uid-710_-8237763529823316207","username":"username-710_6972794678612554028","approvedUser":null},{ "uid":"uid-711_-2583011060046047562","username":"username-711_-4840351723061577086","approvedUser":null},{ "uid":"uid-712_-6873730865061371332","username":"username-712_68211620564959466","approvedUser":null},{ "uid":"uid-713_19760563598375724","username":"username-713_7801971141157651031","approvedUser":null},{ "uid":"uid-714_-3634702827140670497","username":"username-714_-7585589668840641367","approvedUser":null},{ "uid":"uid-715_3970841463757622339","username":"username-715_-174944011341821426","approvedUser":null},{ "uid":"uid-716_8928870054314770142","username":"username-716_9016482382174900133","approvedUser":null},{ "uid":"uid-717_-6733361867744183789","username":"username-717_6002744311108530787","approvedUser":null},{ "uid":"uid-718_-306345974840323676","username":"username-718_2839305524657546810","approvedUser":null},{ "uid":"uid-719_203383599178855544","username":"username-719_-2374427401224115337","approvedUser":null},{ "uid":"uid-720_3081112307506320029","username":"username-720_8145314876673625707","approvedUser":null},{ "uid":"uid-721_6429313621644156815","username":"username-721_1507788325367496013","approvedUser":null},{ "uid":"uid-722_6050824244721087674","username":"username-722_-269722795504347042","approvedUser":null},{ "uid":"uid-723_-6542108310163793859","username":"username-723_-9216309322401161138","approvedUser":null},{ "uid":"uid-724_8817151826837531632","username":"username-724_6751440013322135262","approvedUser":null},{ "uid":"uid-725_-7833876813844363096","username":"username-725_3998983476464526790","approvedUser":null},{ "uid":"uid-726_7473295112598194617","username":"username-726_4552936001469734872","approvedUser":null},{ "uid":"uid-727_8273956927059299515","username":"username-727_5858993689746026596","approvedUser":null},{ "uid":"uid-728_-1562090845420131951","username":"username-728_3947519778861952222","approvedUser":null},{ "uid":"uid-729_-5978863358267365722","username":"username-729_8865578428004629694","approvedUser":null},{ "uid":"uid-730_-6527398427460833926","username":"username-730_-3856752967957591129","approvedUser":null},{ "uid":"uid-731_4930826128701932380","username":"username-731_801829005526907339","approvedUser":null},{ "uid":"uid-732_-5090628997407929736","username":"username-732_-272139073523061107","approvedUser":null},{ "uid":"uid-733_-7366504197528432876","username":"username-733_-5149415599459847833","approvedUser":null},{ "uid":"uid-734_5909101686882477334","username":"username-734_-4234303393132532467","approvedUser":null},{ "uid":"uid-735_-3448927648614442551","username":"username-735_-6653482423405133614","approvedUser":null},{ "uid":"uid-736_-2676258471509852704","username":"username-

736_1923468134633636939","approvedUser":null},{ "uid":"uid-737_-5789986594389275718","username":"username-737_-
8968788447469313743","approvedUser":null},{ "uid":"uid-738_3585290046651543015","username":"username-
738_8160979628990392203","approvedUser":null},{ "uid":"uid-739_-3233537327326991894","username":"username-739_-
7406117689175195636","approvedUser":null},{ "uid":"uid-740_-5597411395572190400","username":"username-
740_4042451492844757177","approvedUser":null},{ "uid":"uid-741_4792115016597154465","username":"username-741_-
9020214908426950706","approvedUser":null},{ "uid":"uid-742_6734450397492055429","username":"username-
742_8022622288805609698","approvedUser":null},{ "uid":"uid-743_6408642590896019945","username":"username-
743_2774093477147310153","approvedUser":null},{ "uid":"uid-744_-5655320940513138094","username":"username-
744_5837209520794624185","approvedUser":null},{ "uid":"uid-745_-371505939206517230","username":"username-745_-
5374645274832638161","approvedUser":null},{ "uid":"uid-746_67954498814370249","username":"username-746_-
67055830709636845","approvedUser":null},{ "uid":"uid-747_-5955474337388893138","username":"username-
747_3857634411518951578","approvedUser":null},{ "uid":"uid-748_3932882788804018959","username":"username-
748_5477542468759247723","approvedUser":null},{ "uid":"uid-749_1576536083462143125","username":"username-
749_1779295198392876933","approvedUser":null},{ "uid":"uid-750_-6595898309250868","username":"username-750_-
66990799609646497","approvedUser":null},{ "uid":"uid-751_-4191340409648869533","username":"username-
751_2761317212502447696","approvedUser":null},{ "uid":"uid-752_5106676213168065967","username":"username-
752_7501000980163113520","approvedUser":null},{ "uid":"uid-753_-314371029821148939","username":"username-
753_7260810612832636736","approvedUser":null},{ "uid":"uid-754_-4748775880469333784","username":"username-754_-
5572914491599793850","approvedUser":null},{ "uid":"uid-755_3427924072792829552","username":"username-755_-
4746452639804068612","approvedUser":null},{ "uid":"uid-756_265205511413144237","username":"username-
756_3089740964032274602","approvedUser":null},{ "uid":"uid-757_-2870516046824134086","username":"username-757_-
3170291019196198127","approvedUser":null},{ "uid":"uid-758_-8662164911203581771","username":"username-758_-
7925947286900425589","approvedUser":null},{ "uid":"uid-759_-3524867313331486593","username":"username-759_-
3130628672624935313","approvedUser":null},{ "uid":"uid-760_-736628605931042112","username":"username-760_-
5484694094742614854","approvedUser":null},{ "uid":"uid-761_2650932392297921453","username":"username-
761_3085731696695381483","approvedUser":null},{ "uid":"uid-762_6313593583111104269","username":"username-762_-
6218024080474893011","approvedUser":null},{ "uid":"uid-763_-115549054830356232","username":"username-763_-
7837121973259543877","approvedUser":null},{ "uid":"uid-764_-8304220186218783007","username":"username-764_-
7369508669465422209","approvedUser":null},{ "uid":"uid-765_5655278210384551127","username":"username-
765_8404247598382358781","approvedUser":null},{ "uid":"uid-766_-667416327795150073","username":"username-
766_5681656807162789486","approvedUser":null},{ "uid":"uid-767_-2967003390893508731","username":"username-
767_6724348439749818416","approvedUser":null},{ "uid":"uid-768_8015321188515874891","username":"username-768_-
5313891662362630052","approvedUser":null},{ "uid":"uid-769_-1853284048191751807","username":"username-
769_3098641250499730360","approvedUser":null},{ "uid":"uid-770_-5455810314760772610","username":"username-
770_3205729483180699088","approvedUser":null},{ "uid":"uid-771_-5526723840736798706","username":"username-771_-
5507600447945669794","approvedUser":null},{ "uid":"uid-772_8709945370539964533","username":"username-772_-
8744362505478748430","approvedUser":null},{ "uid":"uid-773_633546005846753716","username":"username-773_-
4897422842747253810","approvedUser":null},{ "uid":"uid-774_3545280032856113546","username":"username-774_-
1180772303197127918","approvedUser":null},{ "uid":"uid-775_4057219185951998653","username":"username-775_-
6180437945790812392","approvedUser":null},{ "uid":"uid-776_-8991071875212201703","username":"username-
776_322217951142076705","approvedUser":null},{ "uid":"uid-777_-1600627626405054023","username":"username-777_-
6562063565148992208","approvedUser":null},{ "uid":"uid-778_-190439880489882748","username":"username-
778_1624354441270964278","approvedUser":null},{ "uid":"uid-779_7952163740818529274","username":"username-779_-
4209507124678163346","approvedUser":null},{ "uid":"uid-780_6225336090169529392","username":"username-
780_2066894247842603018","approvedUser":null},{ "uid":"uid-781_-5702940568578291175","username":"username-781_-
7654091446715246065","approvedUser":null},{ "uid":"uid-782_218153968591858182","username":"username-782_-
6284735507792897668","approvedUser":null},{ "uid":"uid-783_-4892424408407477137","username":"username-
783_5921378026530402994","approvedUser":null},{ "uid":"uid-784_6814012131459593203","username":"username-
784_2996866762798839567","approvedUser":null},{ "uid":"uid-785_1452284873275740061","username":"username-
785_6017150768767965915","approvedUser":null},{ "uid":"uid-786_-167489321294807704","username":"username-786_-
2569875843450645145","approvedUser":null},{ "uid":"uid-787_-3677726009874261756","username":"username-787_-
8229227987524231309","approvedUser":null},{ "uid":"uid-788_238411198715388634","username":"username-
788_1495575793226150864","approvedUser":null},{ "uid":"uid-789_6789396432133598030","username":"username-

789_5554429128408593990","approvedUser":null},{ "uid":"uid-790_8724507899250044483","username":"username-790_-
1335250834704177993","approvedUser":null},{ "uid":"uid-791_-5666609989755690269","username":"username-
791_3899616395597919875","approvedUser":null},{ "uid":"uid-792_-4772877110983392353","username":"username-
792_423016870406377006","approvedUser":null},{ "uid":"uid-793_-3755801293155626379","username":"username-
793_1794466556197462336","approvedUser":null},{ "uid":"uid-794_1568452867013614960","username":"username-794_-
5914102106529465748","approvedUser":null},{ "uid":"uid-795_-4211660572068214923","username":"username-
795_5913359082611800564","approvedUser":null},{ "uid":"uid-796_5375853926128638267","username":"username-796_-
4354558102445148085","approvedUser":null},{ "uid":"uid-797_8012773529022652059","username":"username-
797_5718575326127304443","approvedUser":null},{ "uid":"uid-798_1139737040555103241","username":"username-798_-
8027874955156501559","approvedUser":null},{ "uid":"uid-799_3170455757070002273","username":"username-
799_5395194286672237078","approvedUser":null},{ "uid":"uid-800_-7719295608677099801","username":"username-
800_3112495492712230535","approvedUser":null},{ "uid":"uid-801_329682767435668873","username":"username-
801_8621707015393534184","approvedUser":null},{ "uid":"uid-802_4967752427881993062","username":"username-
802_3935972039826366541","approvedUser":null},{ "uid":"uid-803_-1704503353558637677","username":"username-803_-
6206771455438054022","approvedUser":null},{ "uid":"uid-804_3935138834710638619","username":"username-
804_7244080866244200324","approvedUser":null},{ "uid":"uid-805_7410119196905234760","username":"username-805_-
305969525966915913","approvedUser":null},{ "uid":"uid-806_3957941761209820448","username":"username-806_-
4394020132235337286","approvedUser":null},{ "uid":"uid-807_-1732309430945914062","username":"username-
807_1146059563406917873","approvedUser":null},{ "uid":"uid-808_-6028511422155298670","username":"username-808_-
5297903024560634846","approvedUser":null},{ "uid":"uid-809_-8108899214087729119","username":"username-
809_6387346155015159386","approvedUser":null},{ "uid":"uid-810_-4078596519404442863","username":"username-
810_8766369292896660497","approvedUser":null},{ "uid":"uid-811_-8858740221745975897","username":"username-
811_4293874255604643724","approvedUser":null},{ "uid":"uid-812_-7468872088341126077","username":"username-
812_2426201592468823938","approvedUser":null},{ "uid":"uid-813_5756854563418813597","username":"username-813_-
23621202334194686","approvedUser":null},{ "uid":"uid-814_-6204121458723819696","username":"username-
814_7981132505997516179","approvedUser":null},{ "uid":"uid-815_5866899224897484188","username":"username-
815_3682924762586833564","approvedUser":null},{ "uid":"uid-816_-5559675890565957785","username":"username-
816_2743976507804720902","approvedUser":null},{ "uid":"uid-817_3519881116014742917","username":"username-817_-
7047977963224628705","approvedUser":null},{ "uid":"uid-818_-3939855647467178952","username":"username-
818_5465146472457971393","approvedUser":null},{ "uid":"uid-819_3865310971738745338","username":"username-819_-
1812883212900165197","approvedUser":null},{ "uid":"uid-820_4891603009574073977","username":"username-
820_6379879498076381774","approvedUser":null},{ "uid":"uid-821_2653194334298611170","username":"username-821_-
4587491772117454493","approvedUser":null},{ "uid":"uid-822_2135991243997168738","username":"username-
822_8005680154368023132","approvedUser":null},{ "uid":"uid-823_3604598397920626153","username":"username-823_-
2189499680648725855","approvedUser":null},{ "uid":"uid-824_4912627575999284784","username":"username-824_-
5964197711079935172","approvedUser":null},{ "uid":"uid-825_-391168391938409219","username":"username-
825_292252604204062911","approvedUser":null},{ "uid":"uid-826_-5837053586704563547","username":"username-826_-
1567859731757613724","approvedUser":null},{ "uid":"uid-827_5619783231282340050","username":"username-827_-
2614790142901126794","approvedUser":null},{ "uid":"uid-828_-9102101764726592634","username":"username-828_-
7195985122624940560","approvedUser":null},{ "uid":"uid-829_-4846636293770140065","username":"username-
829_5859110193844682083","approvedUser":null},{ "uid":"uid-830_-4389962981722801636","username":"username-
830_2722889969801893660","approvedUser":null},{ "uid":"uid-831_-8999330806351479201","username":"username-
831_5011784014313631877","approvedUser":null},{ "uid":"uid-832_7282596863741844404","username":"username-832_-
8904336097223321397","approvedUser":null},{ "uid":"uid-833_7747473530249677265","username":"username-833_-
3711250225556106559","approvedUser":null},{ "uid":"uid-834_8205920869284965919","username":"username-
834_3301755280761454851","approvedUser":null},{ "uid":"uid-835_-5715199003224138119","username":"username-
835_2239551500791443550","approvedUser":null},{ "uid":"uid-836_3815861987567062066","username":"username-836_-
877766728562465681","approvedUser":null},{ "uid":"uid-837_-4798152791811220643","username":"username-
837_8650719689254177417","approvedUser":null},{ "uid":"uid-838_2215280558943382087","username":"username-838_-
1355901431044262180","approvedUser":null},{ "uid":"uid-839_-1950463624453485240","username":"username-839_-
5702382514257670190","approvedUser":null},{ "uid":"uid-840_-861488620642745669","username":"username-840_-
5270004986431779908","approvedUser":null},{ "uid":"uid-841_-3165067637634014390","username":"username-841_-
5022646966395673426","approvedUser":null},{ "uid":"uid-842_-7872889548609547666","username":"username-842_-

2584809908930615642","approvedUser":null},{ "uid":"uid-843_-8595254703781383371","username":"username-843_-
295978634656597005","approvedUser":null},{ "uid":"uid-844_-3964245967936857414","username":"username-844_-
2938745841315914796","approvedUser":null},{ "uid":"uid-845_-1313065447364562466","username":"username-
845_-4355084549631049566","approvedUser":null},{ "uid":"uid-846_-5375461080339090772","username":"username-846_-
1770288122283245405","approvedUser":null},{ "uid":"uid-847_-4243758222029497578","username":"username-847_-
786980702173657072","approvedUser":null},{ "uid":"uid-848_-8121461415178614259","username":"username-848_-
1380956246369440746","approvedUser":null},{ "uid":"uid-849_-9195847628509838510","username":"username-849_-
2895670857260407027","approvedUser":null},{ "uid":"uid-850_-2358907566204866394","username":"username-850_-
7713830002570786988","approvedUser":null},{ "uid":"uid-851_-7508847795995210576","username":"username-851_-
2200660208314778180","approvedUser":null},{ "uid":"uid-852_-7466912000482248584","username":"username-
852_-7432445775461527078","approvedUser":null},{ "uid":"uid-853_-7504827878754472951","username":"username-853_-
8768436849740863894","approvedUser":null},{ "uid":"uid-854_-982050730178802961","username":"username-854_-
6151810805910659206","approvedUser":null},{ "uid":"uid-855_-1385672100393272433","username":"username-
855_-7501808826408398023","approvedUser":null},{ "uid":"uid-856_-8033906926103970930","username":"username-856_-
5537201771055153694","approvedUser":null},{ "uid":"uid-857_-2520351536411741511","username":"username-857_-
8429661263766464922","approvedUser":null},{ "uid":"uid-858_-6162172520957348011","username":"username-858_-
2524114448800341994","approvedUser":null},{ "uid":"uid-859_-5486229749868295953","username":"username-
859_-3368100413835066177","approvedUser":null},{ "uid":"uid-860_-98358158555224638","username":"username-
860_-7171336391730458990","approvedUser":null},{ "uid":"uid-861_-3368701600989251109","username":"username-
861_-5189875016039870004","approvedUser":null},{ "uid":"uid-862_-8760060093204654272","username":"username-
862_-5329871595055407720","approvedUser":null},{ "uid":"uid-863_-5543617697018694526","username":"username-
863_-1587774453904307582","approvedUser":null},{ "uid":"uid-864_-5210903019117246286","username":"username-864_-
4642516659185030872","approvedUser":null},{ "uid":"uid-865_-1147298790991990875","username":"username-
865_-8539697889553191747","approvedUser":null},{ "uid":"uid-866_-5068210450282849833","username":"username-866_-
4219026618207858189","approvedUser":null},{ "uid":"uid-867_-365966513325284032","username":"username-
867_-7684080497232731660","approvedUser":null},{ "uid":"uid-868_-5253983528969158331","username":"username-
868_-6793335421910130611","approvedUser":null},{ "uid":"uid-869_-7052587910762823182","username":"username-869_-
5335096455708759740","approvedUser":null},{ "uid":"uid-870_-9019496194191408384","username":"username-
870_-4318330191873719414","approvedUser":null},{ "uid":"uid-871_-6659382277898956033","username":"username-871_-
6924938542385332803","approvedUser":null},{ "uid":"uid-872_-5036183967881334379","username":"username-872_-
5607719457365706257","approvedUser":null},{ "uid":"uid-873_-5934946590820350668","username":"username-873_-
6369366318685363670","approvedUser":null},{ "uid":"uid-874_-3036633097447871388","username":"username-874_-
5701856497890565644","approvedUser":null},{ "uid":"uid-875_-8869099676680438108","username":"username-
875_-7585945141681140642","approvedUser":null},{ "uid":"uid-876_-5947804948036405562","username":"username-876_-
6105612722920106242","approvedUser":null},{ "uid":"uid-877_-4749159941453539496","username":"username-
877_-799524649218543140","approvedUser":null},{ "uid":"uid-878_-7672584077735192088","username":"username-878_-
7102976858454522708","approvedUser":null},{ "uid":"uid-879_-6843234470099424346","username":"username-
879_-5150919443662259469","approvedUser":null},{ "uid":"uid-880_-5615682149132149277","username":"username-880_-
3183148844569714624","approvedUser":null},{ "uid":"uid-881_-5318425910769897704","username":"username-881_-
7974748286298395741","approvedUser":null},{ "uid":"uid-882_-5426825830399622963","username":"username-882_-
5431871461198317280","approvedUser":null},{ "uid":"uid-883_-7957009529823230201","username":"username-883_-
2428258919825445490","approvedUser":null},{ "uid":"uid-884_-5787755940453697545","username":"username-884_-
8272397151610513066","approvedUser":null},{ "uid":"uid-885_-8558238864063783026","username":"username-
885_-6522790795814811481","approvedUser":null},{ "uid":"uid-886_-53678858480553390","username":"username-
886_-5906960615483323921","approvedUser":null},{ "uid":"uid-887_-1684556805074709033","username":"username-887_-
6332562811231342858","approvedUser":null},{ "uid":"uid-888_-8269761917144510565","username":"username-
888_-5875826683618029602","approvedUser":null},{ "uid":"uid-889_-5113990807545768623","username":"username-
889_-6086545033224993847","approvedUser":null},{ "uid":"uid-890_-477574293856362882","username":"username-
890_-1702791304174888064","approvedUser":null},{ "uid":"uid-891_-6682876177152891952","username":"username-891_-
2582215649075715754","approvedUser":null},{ "uid":"uid-892_-681330050070814737","username":"username-
892_-2271567797965652676","approvedUser":null},{ "uid":"uid-893_-2615135690914312212","username":"username-893_-
1297579413114723474","approvedUser":null},{ "uid":"uid-894_-8656615249292894859","username":"username-894_-
9161482516872171602","approvedUser":null},{ "uid":"uid-895_-6948035457419664962","username":"username-895_-

7173201823750520065","approvedUser":null},{ "uid":"uid-896_-8418129637612400671","username":"username-896_-8427126334563885470","approvedUser":null},{ "uid":"uid-897_-7092614010377765670","username":"username-897_-3125581636370938770","approvedUser":null},{ "uid":"uid-898_-5575788934196515452","username":"username-898_-7931786552700130587","approvedUser":null},{ "uid":"uid-899_-9179487561509967257","username":"username-899_-4581894664447473534","approvedUser":null},{ "uid":"uid-900_-4256281735756462516","username":"username-900_-9034396163583834960","approvedUser":null},{ "uid":"uid-901_-5518937309978695927","username":"username-901_-5659451021016124184","approvedUser":null},{ "uid":"uid-902_-5724659273619939657","username":"username-902_-1204120520806861686","approvedUser":null},{ "uid":"uid-903_-8618027174290785879","username":"username-903_-3368661292431053647","approvedUser":null},{ "uid":"uid-904_-6836368458047629088","username":"username-904_-6460961997100110011","approvedUser":null},{ "uid":"uid-905_-576177549822969823","username":"username-905_-1890611499577792719","approvedUser":null},{ "uid":"uid-906_-6300007257757868188","username":"username-906_-3314363058958819334","approvedUser":null},{ "uid":"uid-907_-2192308472356855307","username":"username-907_-935263973911838875","approvedUser":null},{ "uid":"uid-908_-6388500708703568416","username":"username-908_-6523330424037190911","approvedUser":null},{ "uid":"uid-909_-4311827137701484079","username":"username-909_-314262190288046418","approvedUser":null},{ "uid":"uid-910_-420547575720952938","username":"username-910_-5218779976138380370","approvedUser":null},{ "uid":"uid-911_-4797994536827529750","username":"username-911_-8849849560955323501","approvedUser":null},{ "uid":"uid-912_-5483206063032261397","username":"username-912_-6937561870293484158","approvedUser":null},{ "uid":"uid-913_-423251972892264764","username":"username-913_-9105731600532229662","approvedUser":null},{ "uid":"uid-914_-5694619586986359440","username":"username-914_-5114660498119197147","approvedUser":null},{ "uid":"uid-915_-793011801942837472","username":"username-915_-5875021737226745047","approvedUser":null},{ "uid":"uid-916_-8693914955764340229","username":"username-916_-7276622598814702112","approvedUser":null},{ "uid":"uid-917_-727201718547557723","username":"username-917_-8976960348228467150","approvedUser":null},{ "uid":"uid-918_-3373784354013752868","username":"username-918_-2501745373912981471","approvedUser":null},{ "uid":"uid-919_-9218489079874217663","username":"username-919_-748353430309706962","approvedUser":null},{ "uid":"uid-920_-3864274675984907272","username":"username-920_-7278416771402707973","approvedUser":null},{ "uid":"uid-921_-1908891486544553757","username":"username-921_-3967102119587048497","approvedUser":null},{ "uid":"uid-922_-958606161654300886","username":"username-922_-1805566268410966877","approvedUser":null},{ "uid":"uid-923_-1376944986447732175","username":"username-923_-3418866991964687795","approvedUser":null},{ "uid":"uid-924_-1600887626930018803","username":"username-924_-4874045853830392308","approvedUser":null},{ "uid":"uid-925_-2239820604546918989","username":"username-925_-1344873584300703504","approvedUser":null},{ "uid":"uid-926_-1545783101755981499","username":"username-926_-4496222611822510316","approvedUser":null},{ "uid":"uid-927_-2195858000077116190","username":"username-927_-548488638550099191","approvedUser":null},{ "uid":"uid-928_-7822931129025946433","username":"username-928_-9193488058055754797","approvedUser":null},{ "uid":"uid-929_-4000016919813141554","username":"username-929_-9202461322151364674","approvedUser":null},{ "uid":"uid-930_-855679071680203216","username":"username-930_-182167604361065347","approvedUser":null},{ "uid":"uid-931_-5237782086880239472","username":"username-931_-6394029503899505090","approvedUser":null},{ "uid":"uid-932_-3428434238605618974","username":"username-932_-9057899458774615738","approvedUser":null},{ "uid":"uid-933_-6303279436602663775","username":"username-933_-395170472504938357","approvedUser":null},{ "uid":"uid-934_-3093416327015708451","username":"username-934_-7025573464248197398","approvedUser":null},{ "uid":"uid-935_-8934986111602719056","username":"username-935_-4800565926016581293","approvedUser":null},{ "uid":"uid-936_-6106193619966847651","username":"username-936_-812387257420583456","approvedUser":null},{ "uid":"uid-937_-5992812823381230261","username":"username-937_-5750504498742094493","approvedUser":null},{ "uid":"uid-938_-5441574512668863340","username":"username-938_-2943578424481374906","approvedUser":null},{ "uid":"uid-939_-5399901772831076134","username":"username-939_-4705772170615438115","approvedUser":null},{ "uid":"uid-940_-7830367868781068790","username":"username-940_-1093578794892939741","approvedUser":null},{ "uid":"uid-941_-6321827333814085815","username":"username-941_-8725205384582230551","approvedUser":null},{ "uid":"uid-942_-1616717983011904220","username":"username-942_-6851734708861372504","approvedUser":null},{ "uid":"uid-943_-6292682400081247131","username":"username-943_-2558726802681229918","approvedUser":null},{ "uid":"uid-944_-8079302446345572643","username":"username-944_-3163520732968666590","approvedUser":null},{ "uid":"uid-945_-3977235947983624784","username":"username-945_-5938771599752373301","approvedUser":null},{ "uid":"uid-946_-4229002077908414117","username":"username-946_-7604155955365845091","approvedUser":null},{ "uid":"uid-947_-3167357411094293095","username":"username-947_-4934885667279505267","approvedUser":null},{ "uid":"uid-948_-6510140137774309539","username":"username-948_-

8313294066407549502","approvedUser":null},{ "uid":"uid-949_1214748778623876926","username":"username-949_-
402189804140759821","approvedUser":null},{ "uid":"uid-950_-4128729017455207802","username":"username-
950_3666279773990526890","approvedUser":null},{ "uid":"uid-951_5215564143857221705","username":"username-
951_4757415195665345391","approvedUser":null},{ "uid":"uid-952_-4879201736013244067","username":"username-
952_2507777273561518956","approvedUser":null},{ "uid":"uid-953_8571562930311713961","username":"username-
953_7016163393571358803","approvedUser":null},{ "uid":"uid-954_-1933058302645677150","username":"username-954_-
372528380065768833","approvedUser":null},{ "uid":"uid-955_2449084042599226131","username":"username-
955_6233388787742204505","approvedUser":null},{ "uid":"uid-956_-7986833205478585316","username":"username-
956_6418494249285929725","approvedUser":null},{ "uid":"uid-957_5368866350076324220","username":"username-957_-
4435991953948482788","approvedUser":null},{ "uid":"uid-958_384520652564786450","username":"username-958_-
8162691492804962269","approvedUser":null},{ "uid":"uid-959_-2133891601219194464","username":"username-959_-
5701078751464882276","approvedUser":null},{ "uid":"uid-960_-2344856472161618734","username":"username-960_-
6542393836968078635","approvedUser":null},{ "uid":"uid-961_-5803158411551574657","username":"username-
961_8257923033433047230","approvedUser":null},{ "uid":"uid-962_-193239950130908627","username":"username-
962_6241110247524137495","approvedUser":null},{ "uid":"uid-963_-788391963555738662","username":"username-963_-
7466278638235870262","approvedUser":null},{ "uid":"uid-964_1022263725178327289","username":"username-964_-
7068026292135601259","approvedUser":null},{ "uid":"uid-965_1265341768081601722","username":"username-
965_1435621930956135147","approvedUser":null},{ "uid":"uid-966_-2471735723588127090","username":"username-966_-
5836927179160878052","approvedUser":null},{ "uid":"uid-967_8871926911845203434","username":"username-
967_2533588914522461619","approvedUser":null},{ "uid":"uid-968_4090381658403696202","username":"username-
968_8766639823539339294","approvedUser":null},{ "uid":"uid-969_3145471527081211768","username":"username-969_-
1017626836501907052","approvedUser":null},{ "uid":"uid-970_460483834800875021","username":"username-
970_6192308297583193139","approvedUser":null},{ "uid":"uid-971_-7380252105409562568","username":"username-971_-
3706210197383929240","approvedUser":null},{ "uid":"uid-972_-3497974513413555523","username":"username-
972_1977189429328500114","approvedUser":null},{ "uid":"uid-973_-3448878380671847871","username":"username-
973_6259980741650301673","approvedUser":null},{ "uid":"uid-974_8781179919243695300","username":"username-974_-
5715649405756167524","approvedUser":null},{ "uid":"uid-975_1667219432083953517","username":"username-
975_8586979543319338297","approvedUser":null},{ "uid":"uid-976_7826396042370356365","username":"username-
976_1329191154663439186","approvedUser":null},{ "uid":"uid-977_-2346193956526420515","username":"username-977_-
8071344750219949220","approvedUser":null},{ "uid":"uid-978_-6076127113109041398","username":"username-978_-
1245077927954298406","approvedUser":null},{ "uid":"uid-979_6242133924194301043","username":"username-
979_3419177384295749562","approvedUser":null},{ "uid":"uid-980_-4916679281153828303","username":"username-
980_3586086860559657726","approvedUser":null},{ "uid":"uid-981_1843509545711023959","username":"username-981_-
7790626131363043938","approvedUser":null},{ "uid":"uid-982_5141682374575237498","username":"username-
982_5714916154519221932","approvedUser":null},{ "uid":"uid-983_-362085163262987149","username":"username-
983_4915718421499128249","approvedUser":null},{ "uid":"uid-984_-8104053554723194269","username":"username-984_-
6558632878526085001","approvedUser":null},{ "uid":"uid-985_-5199828430822789129","username":"username-985_-
9082355618973601121","approvedUser":null},{ "uid":"uid-986_-6010669299046002452","username":"username-
986_1039089908708806676","approvedUser":null},{ "uid":"uid-987_-4484092251809539883","username":"username-987_-
8523382748929228472","approvedUser":null},{ "uid":"uid-988_-8982050197939665892","username":"username-988_-
6432556701131836962","approvedUser":null},{ "uid":"uid-989_-328345648383664826","username":"username-
989_797452547777797521","approvedUser":null},{ "uid":"uid-990_2143179230326823546","username":"username-990_-
4743860218697664685","approvedUser":null},{ "uid":"uid-991_7018758412073242435","username":"username-991_-
2563589721452531015","approvedUser":null},{ "uid":"uid-992_-4176579757859260780","username":"username-
992_4485080687665631125","approvedUser":null},{ "uid":"uid-993_2191776148148427020","username":"username-
993_3608958190533300196","approvedUser":null},{ "uid":"uid-994_-2468207764255763589","username":"username-
994_1273704360945803242","approvedUser":null},{ "uid":"uid-995_8631626521119949221","username":"username-
995_4078258630135564443","approvedUser":null},{ "uid":"uid-996_-1570272270136258235","username":"username-996_-
628315177634982851","approvedUser":null},{ "uid":"uid-997_6012046471971619156","username":"username-
997_3287144557088651594","approvedUser":null},{ "uid":"uid-998_-5026474191518215798","username":"username-
998_6803756188442550170","approvedUser":null},{ "uid":"uid-999_-2979865592950086107","username":"username-999_-
326225470764870719","approvedUser":null},{ "uid":"uid-1000_-2776490639246072856","username":"username-
1000_6066006090985512118","approvedUser":null},{ "uid":"uid-1001_-4482919973241632666","username":"username-

1001_4466168266764642228","approvedUser":null},{ "uid":"uid-1002_-4675318616917114240","username":"username-1002_6412275051869699684","approvedUser":null},{ "uid":"uid-1003_-5015525270907842773","username":"username-1003_2535909859762289467","approvedUser":null},{ "uid":"uid-1004_5150899079373037174","username":"username-1004_-8141407390716075999","approvedUser":null},{ "uid":"uid-1005_-6166278139634694965","username":"username-1005_5643989041215664979","approvedUser":null},{ "uid":"uid-1006_1176467963390426228","username":"username-1006_-3291915882622024055","approvedUser":null},{ "uid":"uid-1007_-5145260104788362399","username":"username-1007_-6742115890443662246","approvedUser":null},{ "uid":"uid-1008_4744810524546244903","username":"username-1008_6062569838436425695","approvedUser":null},{ "uid":"uid-1009_2527232667367407948","username":"username-1009_4727538707223831191","approvedUser":null},{ "uid":"uid-1010_2815821942290206145","username":"username-1010_8818113363119854730","approvedUser":null},{ "uid":"uid-1011_8284918539789804975","username":"username-1011_2070720249220975280","approvedUser":null},{ "uid":"uid-1012_-5058790289469535114","username":"username-1012_-3036464064243243118","approvedUser":null},{ "uid":"uid-1013_1471194542055329009","username":"username-1013_2819773477055248945","approvedUser":null},{ "uid":"uid-1014_-1202484042367546113","username":"username-1014_-1346376576573937881","approvedUser":null},{ "uid":"uid-1015_6633065012026791231","username":"username-1015_-6878911590337921012","approvedUser":null},{ "uid":"uid-1016_2646839411369742618","username":"username-1016_-5891811666857792230","approvedUser":null},{ "uid":"uid-1017_7522043006458546949","username":"username-1017_-1202009082168734444","approvedUser":null},{ "uid":"uid-1018_-9040534278258054905","username":"username-1018_-8067324504634167454","approvedUser":null},{ "uid":"uid-1019_1895435472529157497","username":"username-1019_3739582875147816064","approvedUser":null},{ "uid":"uid-1020_-8027224952925093876","username":"username-1020_-2794606032124567399","approvedUser":null},{ "uid":"uid-1021_-7673367807965789424","username":"username-1021_1560894133733657711","approvedUser":null},{ "uid":"uid-1022_9005294378304534771","username":"username-1022_92103265175330941","approvedUser":null},{ "uid":"uid-1023_6265236391698972167","username":"username-1023_5257063526995474014","approvedUser":null},{ "uid":"uid-1024_-3958496334700356985","username":"username-1024_-2462202758230590353","approvedUser":null},{ "uid":"uid-1025_-841495767785475877","username":"username-1025_-800148548219758669","approvedUser":null},{ "uid":"uid-1026_3028970804821478427","username":"username-1026_2597180077664796663","approvedUser":null},{ "uid":"uid-1027_379942574718505681","username":"username-1027_8002791424647428662","approvedUser":null},{ "uid":"uid-1028_7676908406341001730","username":"username-1028_-3800483850787334850","approvedUser":null},{ "uid":"uid-1029_-8366519227699178385","username":"username-1029_-5017817569824355182","approvedUser":null},{ "uid":"uid-1030_370346810258307608","username":"username-1030_620955240064578962","approvedUser":null},{ "uid":"uid-1031_8039339920460252161","username":"username-1031_-3054173713470924071","approvedUser":null},{ "uid":"uid-1032_1710102708824119","username":"username-1032_-1232372545044518426","approvedUser":null},{ "uid":"uid-1033_4784548512989544981","username":"username-1033_3285058319322158845","approvedUser":null},{ "uid":"uid-1034_5464900249545568963","username":"username-1034_4102967026408086351","approvedUser":null},{ "uid":"uid-1035_-7593491578546158081","username":"username-1035_8475858613154541912","approvedUser":null},{ "uid":"uid-1036_-8589998132959240417","username":"username-1036_7520315969398043528","approvedUser":null},{ "uid":"uid-1037_-7003923796526337450","username":"username-1037_-4535988635224555227","approvedUser":null},{ "uid":"uid-1038_-7040351241992712650","username":"username-1038_3276166400203640175","approvedUser":null},{ "uid":"uid-1039_-5708078147235415691","username":"username-1039_-651451520701121276","approvedUser":null},{ "uid":"uid-1040_207222384241387763","username":"username-1040_32760875583239558","approvedUser":null},{ "uid":"uid-1041_-8605804877598404101","username":"username-1041_-6834081836853853318","approvedUser":null},{ "uid":"uid-1042_-1509252718216225592","username":"username-1042_-1190684201713862030","approvedUser":null},{ "uid":"uid-1043_-3051421215062066357","username":"username-1043_-6464387551319280786","approvedUser":null},{ "uid":"uid-1044_-2923983599719077845","username":"username-1044_-1726017583518904960","approvedUser":null},{ "uid":"uid-1045_-5583247464000810483","username":"username-1045_-5300825607799040850","approvedUser":null},{ "uid":"uid-1046_2535319432653502229","username":"username-1046_3303116601840153371","approvedUser":null},{ "uid":"uid-1047_2541945564571022507","username":"username-1047_1207406047950131304","approvedUser":null},{ "uid":"uid-1048_5031605393426714497","username":"username-1048_4898165124269638487","approvedUser":null},{ "uid":"uid-1049_-7084398526456464541","username":"username-1049_-3465469652359595599","approvedUser":null},{ "uid":"uid-1050_8338033494648016371","username":"username-1050_5276981196270729834","approvedUser":null},{ "uid":"uid-1051_-1586561244632290563","username":"username-1051_1897430387133556226","approvedUser":null},{ "uid":"uid-1052_3079887112311534062","username":"username-1052_-6135970770255324883","approvedUser":null},{ "uid":"uid-1053_7013121992276520156","username":"username-1053_7686861320175244725","approvedUser":null},{ "uid":"uid-1054_101601028912675453","username":"username-

1054_2075516043486107039","approvedUser":null},{ "uid":"uid-1055_-6070049612426211349","username":"username-1055_-6638027487250191220","approvedUser":null},{ "uid":"uid-1056_-462055622235536725","username":"username-1056_-446444585695061150","approvedUser":null},{ "uid":"uid-1057_-6559956883852603669","username":"username-1057_-376667879757749973","approvedUser":null},{ "uid":"uid-1058_-7197640214703869451","username":"username-1058_-8625460739281380292","approvedUser":null},{ "uid":"uid-1059_-271043689483611810","username":"username-1059_-1579310442306572953","approvedUser":null},{ "uid":"uid-1060_-3577770087603041584","username":"username-1060_-2684916698863644824","approvedUser":null},{ "uid":"uid-1061_-8755410719742066699","username":"username-1061_-6982918493087200963","approvedUser":null},{ "uid":"uid-1062_-6703942046511964726","username":"username-1062_-45583033916065794","approvedUser":null},{ "uid":"uid-1063_-8198869515875243650","username":"username-1063_-7284832164005891321","approvedUser":null},{ "uid":"uid-1064_-2858135416883604792","username":"username-1064_-1544312070812374609","approvedUser":null},{ "uid":"uid-1065_-109691500256659835","username":"username-1065_-7052034404457715345","approvedUser":null},{ "uid":"uid-1066_-8518326024299681577","username":"username-1066_-6058155856656349965","approvedUser":null},{ "uid":"uid-1067_-302205485037097292","username":"username-1067_-7304521078655919806","approvedUser":null},{ "uid":"uid-1068_-7134122831265222982","username":"username-1068_-8671953144383696440","approvedUser":null},{ "uid":"uid-1069_-21174965197147362","username":"username-1069_-3762561740342112672","approvedUser":null},{ "uid":"uid-1070_-662163316650982408","username":"username-1070_-3604422034690064050","approvedUser":null},{ "uid":"uid-1071_-1593615837462520746","username":"username-1071_-9160911732339273417","approvedUser":null},{ "uid":"uid-1072_-6533761405569671697","username":"username-1072_-2479239947398431163","approvedUser":null},{ "uid":"uid-1073_-242768822916034069","username":"username-1073_-8885138973670445062","approvedUser":null},{ "uid":"uid-1074_-6258309262895558177","username":"username-1074_-8487889843279069804","approvedUser":null},{ "uid":"uid-1075_-3923815166171953335","username":"username-1075_-2844343954973455350","approvedUser":null},{ "uid":"uid-1076_-294776470593233070","username":"username-1076_-6689519565061284154","approvedUser":null},{ "uid":"uid-1077_-4170072981570361513","username":"username-1077_-7564623198854408429","approvedUser":null},{ "uid":"uid-1078_-3050783844170032709","username":"username-1078_-461406557787169928","approvedUser":null},{ "uid":"uid-1079_-2043511536761117511","username":"username-1079_-8350259748301128499","approvedUser":null},{ "uid":"uid-1080_-8743602668831418266","username":"username-1080_-8335757937665766099","approvedUser":null},{ "uid":"uid-1081_-2404827608564496283","username":"username-1081_-4764983760847100536","approvedUser":null},{ "uid":"uid-1082_-4044228004820340304","username":"username-1082_-484592988438328106","approvedUser":null},{ "uid":"uid-1083_-5153889461885543267","username":"username-1083_-8881069047245246516","approvedUser":null},{ "uid":"uid-1084_-3661780898036560959","username":"username-1084_-8385133855636644454","approvedUser":null},{ "uid":"uid-1085_-3915320602342665385","username":"username-1085_-149451367544721563","approvedUser":null},{ "uid":"uid-1086_-5655744459640827624","username":"username-1086_-986557134247303354","approvedUser":null},{ "uid":"uid-1087_-6767649972234426097","username":"username-1087_-6770484775531923644","approvedUser":null},{ "uid":"uid-1088_-1757587996939260160","username":"username-1088_-7875463954815111570","approvedUser":null},{ "uid":"uid-1089_-8074126717686643396","username":"username-1089_-3947176679425730228","approvedUser":null},{ "uid":"uid-1090_-7067669907993348940","username":"username-1090_-1857127601496852805","approvedUser":null},{ "uid":"uid-1091_-3621739726617079988","username":"username-1091_-3187793324795333690","approvedUser":null},{ "uid":"uid-1092_-5015125349062360272","username":"username-1092_-8539067958984144971","approvedUser":null},{ "uid":"uid-1093_-7182643410056779304","username":"username-1093_-3383113901749131741","approvedUser":null},{ "uid":"uid-1094_-2497584966673202149","username":"username-1094_-308654580946446100","approvedUser":null},{ "uid":"uid-1095_-3872992668972275556","username":"username-1095_-3378336209736420387","approvedUser":null},{ "uid":"uid-1096_-4338045655685489897","username":"username-1096_-2546398149263581679","approvedUser":null},{ "uid":"uid-1097_-5992230689514457162","username":"username-1097_-8718613947273937163","approvedUser":null},{ "uid":"uid-1098_-4233465163226327363","username":"username-1098_-4663242217737061041","approvedUser":null},{ "uid":"uid-1099_-8946560584431576246","username":"username-1099_-6169299558392414868","approvedUser":null},{ "uid":"uid-1100_-5425379642261307205","username":"username-1100_-6250492932354456801","approvedUser":null},{ "uid":"uid-1101_-6574319751718574481","username":"username-1101_-8950674592680066308","approvedUser":null},{ "uid":"uid-1102_-7889851446544779285","username":"username-1102_-5469487357174621221","approvedUser":null},{ "uid":"uid-1103_-8623281488704664576","username":"username-1103_-1748570509892080909","approvedUser":null},{ "uid":"uid-1104_-7595130455966085661","username":"username-1104_-1184898413536246813","approvedUser":null},{ "uid":"uid-1105_-5460418473914290973","username":"username-1105_-5572028562907485636","approvedUser":null},{ "uid":"uid-1106_-5965455914775786443","username":"username-1106_-8649972566584811537","approvedUser":null},{ "uid":"uid-1107_-2296680215204633511","username":"username-

1107_6637352615375040071","approvedUser":null},{\"uid\":\"uid-1108_2745296802971526122\",\"username\":\"username-1108_-5349882997153260186\",\"approvedUser\":null},{\"uid\":\"uid-1109_7875627452909056711\",\"username\":\"username-1109_-7417600891361172638\",\"approvedUser\":null},{\"uid\":\"uid-1110_-1188128819761109261\",\"username\":\"username-1110_666259058557019502\",\"approvedUser\":null},{\"uid\":\"uid-1111_-903243675819477145\",\"username\":\"username-1111_8020604336586123067\",\"approvedUser\":null},{\"uid\":\"uid-1112_7665177129803844452\",\"username\":\"username-1112_8508524296614808731\",\"approvedUser\":null},{\"uid\":\"uid-1113_860903458107477687\",\"username\":\"username-1113_-8910694638777855641\",\"approvedUser\":null},{\"uid\":\"uid-1114_7628156901140219454\",\"username\":\"username-1114_4644814373397961635\",\"approvedUser\":null},{\"uid\":\"uid-1115_-8504142852298364758\",\"username\":\"username-1115_7302041902901201887\",\"approvedUser\":null},{\"uid\":\"uid-1116_7271102744126242693\",\"username\":\"username-1116_5281443502714023903\",\"approvedUser\":null},{\"uid\":\"uid-1117_7120125910941553782\",\"username\":\"username-1117_7247220146046794622\",\"approvedUser\":null},{\"uid\":\"uid-1118_2580593226604162756\",\"username\":\"username-1118_-5685800719819330609\",\"approvedUser\":null},{\"uid\":\"uid-1119_7553579532141520152\",\"username\":\"username-1119_2863939522288318596\",\"approvedUser\":null},{\"uid\":\"uid-1120_4547082759376537478\",\"username\":\"username-1120_-2610046692120184941\",\"approvedUser\":null},{\"uid\":\"uid-1121_-1677443829315508786\",\"username\":\"username-1121_3024854930279208726\",\"approvedUser\":null},{\"uid\":\"uid-1122_-5153299958791506466\",\"username\":\"username-1122_4238569937399107954\",\"approvedUser\":null},{\"uid\":\"uid-1123_-1934510571171045734\",\"username\":\"username-1123_4873831076985143227\",\"approvedUser\":null},{\"uid\":\"uid-1124_8647648516340116192\",\"username\":\"username-1124_-4152001395785585065\",\"approvedUser\":null},{\"uid\":\"uid-1125_190341385336154310\",\"username\":\"username-1125_-4529123809603086368\",\"approvedUser\":null},{\"uid\":\"uid-1126_-2246658470370363099\",\"username\":\"username-1126_-5092839124617319857\",\"approvedUser\":null},{\"uid\":\"uid-1127_-961693039031446693\",\"username\":\"username-1127_-5664381478559106474\",\"approvedUser\":null},{\"uid\":\"uid-1128_146055711641518411\",\"username\":\"username-1128_-623566297204029172\",\"approvedUser\":null},{\"uid\":\"uid-1129_5918447737334749096\",\"username\":\"username-1129_-2629375976242020401\",\"approvedUser\":null},{\"uid\":\"uid-1130_2422051020298464883\",\"username\":\"username-1130_9191394346833790565\",\"approvedUser\":null},{\"uid\":\"uid-1131_9002792584878611905\",\"username\":\"username-1131_4256909965899208548\",\"approvedUser\":null},{\"uid\":\"uid-1132_-9190334930464947519\",\"username\":\"username-1132_8157385024927987412\",\"approvedUser\":null},{\"uid\":\"uid-1133_-3682480038517550851\",\"username\":\"username-1133_-505300757615366382\",\"approvedUser\":null},{\"uid\":\"uid-1134_4598891549490164704\",\"username\":\"username-1134_8322462474790582262\",\"approvedUser\":null},{\"uid\":\"uid-1135_1764510780516009080\",\"username\":\"username-1135_-7595596187561792456\",\"approvedUser\":null},{\"uid\":\"uid-1136_4748103014479828859\",\"username\":\"username-1136_-4280226017371672305\",\"approvedUser\":null},{\"uid\":\"uid-1137_746968553487804303\",\"username\":\"username-1137_7481843409981666330\",\"approvedUser\":null},{\"uid\":\"uid-1138_-5169355093877596219\",\"username\":\"username-1138_-3507775360171957373\",\"approvedUser\":null},{\"uid\":\"uid-1139_1742243632522037862\",\"username\":\"username-1139_-8612439809246722975\",\"approvedUser\":null},{\"uid\":\"uid-1140_-3141069407059023886\",\"username\":\"username-1140_-1580544413336663202\",\"approvedUser\":null},{\"uid\":\"uid-1141_8390063715616092773\",\"username\":\"username-1141_3436269850473929625\",\"approvedUser\":null},{\"uid\":\"uid-1142_-5218495253548003140\",\"username\":\"username-1142_-3672463719531982899\",\"approvedUser\":null},{\"uid\":\"uid-1143_6081560183042047671\",\"username\":\"username-1143_-6595180973992103502\",\"approvedUser\":null},{\"uid\":\"uid-1144_5808396133967326431\",\"username\":\"username-1144_6973768590989165475\",\"approvedUser\":null},{\"uid\":\"uid-1145_5203102746315234506\",\"username\":\"username-1145_2079194704877322368\",\"approvedUser\":null},{\"uid\":\"uid-1146_5735206545796464028\",\"username\":\"username-1146_-4497866381069470251\",\"approvedUser\":null},{\"uid\":\"uid-1147_3511335325581511889\",\"username\":\"username-1147_-9170998016394830789\",\"approvedUser\":null},{\"uid\":\"uid-1148_-5153694404841078982\",\"username\":\"username-1148_-5241978645835460098\",\"approvedUser\":null},{\"uid\":\"uid-1149_-7792709901860145595\",\"username\":\"username-1149_6971121757412006912\",\"approvedUser\":null},{\"uid\":\"uid-1150_-1806369991726774911\",\"username\":\"username-1150_7808628767245635804\",\"approvedUser\":null},{\"uid\":\"uid-1151_-5965439526438040480\",\"username\":\"username-1151_8124410956010033741\",\"approvedUser\":null},{\"uid\":\"uid-1152_510607397114095677\",\"username\":\"username-1152_-2091989832706197295\",\"approvedUser\":null},{\"uid\":\"uid-1153_-7022392122655915360\",\"username\":\"username-1153_8974514089449723273\",\"approvedUser\":null},{\"uid\":\"uid-1154_2307619387586281409\",\"username\":\"username-1154_-9203513748680380829\",\"approvedUser\":null},{\"uid\":\"uid-1155_-8056549259441893046\",\"username\":\"username-1155_2093310189960337530\",\"approvedUser\":null},{\"uid\":\"uid-1156_-3596134562171342495\",\"username\":\"username-1156_863655952310752444\",\"approvedUser\":null},{\"uid\":\"uid-1157_7454408167907609061\",\"username\":\"username-1157_7717474902987789004\",\"approvedUser\":null},{\"uid\":\"uid-1158_-5447428153950647306\",\"username\":\"username-1158_-6117163681393263062\",\"approvedUser\":null},{\"uid\":\"uid-1159_-739460727265461317\",\"username\":\"username-1159_-8323489079149299311\",\"approvedUser\":null},{\"uid\":\"uid-1160_9086868855080694878\",\"username\":\"username-1160_-

5656914202000708448","approvedUser":null,{"uid":"uid-1161_2942353510544304219","username":"username-1161_4537257817618712508","approvedUser":null,{"uid":"uid-1162_96048645574812154","username":"username-1162_6882537069075275212","approvedUser":null,{"uid":"uid-1163_-3340632398254831591","username":"username-1163_8381212296183125669","approvedUser":null,{"uid":"uid-1164_-5271355402831156433","username":"username-1164_5499105518472053151","approvedUser":null,{"uid":"uid-1165_879529262583488141","username":"username-1165_6218910796530891128","approvedUser":null,{"uid":"uid-1166_4812866739943514597","username":"username-1166_1353621393667014976","approvedUser":null,{"uid":"uid-1167_3530067030413893392","username":"username-1167_2023483271080102139","approvedUser":null,{"uid":"uid-1168_4334749266610782219","username":"username-1168_3144584270432346343","approvedUser":null,{"uid":"uid-1169_8976588696124760386","username":"username-1169_2400215205792288673","approvedUser":null,{"uid":"uid-1170_-5062166630716791791","username":"username-1170_7487583482054063536","approvedUser":null,{"uid":"uid-1171_5212394832038929799","username":"username-1171_6823835786901252028","approvedUser":null,{"uid":"uid-1172_7842525862459743596","username":"username-1172_3599788078873777489","approvedUser":null,{"uid":"uid-1173_-4340664708580775547","username":"username-1173_7422900765989576262","approvedUser":null,{"uid":"uid-1174_4236114284816812309","username":"username-1174_3087281242714235483","approvedUser":null,{"uid":"uid-1175_-7247666862363636569","username":"username-1175_3803998963580305870","approvedUser":null,{"uid":"uid-1176_6362294194611584523","username":"username-1176_4276988341570472893","approvedUser":null,{"uid":"uid-1177_-2320631697376623459","username":"username-1177_139703807120722370","approvedUser":null,{"uid":"uid-1178_-2157319754937316916","username":"username-1178_5795619954039653312","approvedUser":null,{"uid":"uid-1179_-3784837455798509094","username":"username-1179_7637180092977036576","approvedUser":null,{"uid":"uid-1180_-3456026559422546720","username":"username-1180_6242465305908659702","approvedUser":null,{"uid":"uid-1181_-4799498137280814059","username":"username-1181_684761227606946538","approvedUser":null,{"uid":"uid-1182_-1616971524236851378","username":"username-1182_3093543199784531486","approvedUser":null,{"uid":"uid-1183_-2800391061050712376","username":"username-1183_4054060634814801568","approvedUser":null,{"uid":"uid-1184_1296363893057692965","username":"username-1184_4101806371481941031","approvedUser":null,{"uid":"uid-1185_4937093144230391220","username":"username-1185_3932196435595544923","approvedUser":null,{"uid":"uid-1186_-2397909328317719819","username":"username-1186_6195847637722632076","approvedUser":null,{"uid":"uid-1187_-1809026861959415461","username":"username-1187_6625826188716122113","approvedUser":null,{"uid":"uid-1188_-1683624530207227309","username":"username-1188_3017626811297305222","approvedUser":null,{"uid":"uid-1189_-2018126235823535227","username":"username-1189_6872669072898796705","approvedUser":null,{"uid":"uid-1190_-5450080018991798695","username":"username-1190_7138745237948901761","approvedUser":null,{"uid":"uid-1191_-3952738788933307460","username":"username-1191_6233043397004530730","approvedUser":null,{"uid":"uid-1192_6772290246218051619","username":"username-1192_7975462196426214093","approvedUser":null,{"uid":"uid-1193_-8557672569324897329","username":"username-1193_6873718336350111022","approvedUser":null,{"uid":"uid-1194_7922345605647229779","username":"username-1194_2667579618045530811","approvedUser":null,{"uid":"uid-1195_-1094355267601568685","username":"username-1195_4713243622433168686","approvedUser":null,{"uid":"uid-1196_6801021906474855053","username":"username-1196_2374612166307897533","approvedUser":null,{"uid":"uid-1197_3533556469439275293","username":"username-1197_6837347083663779175","approvedUser":null,{"uid":"uid-1198_7000190098875577340","username":"username-1198_1893905761597195645","approvedUser":null,{"uid":"uid-1199_9047337292542356937","username":"username-1199_4154530981165231452","approvedUser":null,{"uid":"uid-1200_-1719680410209198645","username":"username-1200_8122727118953023548","approvedUser":null,{"uid":"uid-1201_9109123347707597366","username":"username-1201_894419984393822510","approvedUser":null,{"uid":"uid-1202_5390235430214890157","username":"username-1202_1388039509830162006","approvedUser":null,{"uid":"uid-1203_-8895619841800405240","username":"username-1203_8341221425873030128","approvedUser":null,{"uid":"uid-1204_-6560284503843473834","username":"username-1204_6581673870730902874","approvedUser":null,{"uid":"uid-1205_329551442926840357","username":"username-1205_431847337787774915","approvedUser":null,{"uid":"uid-1206_-5466256146168823296","username":"username-1206_1842541981397389558","approvedUser":null,{"uid":"uid-1207_-2822531532768922823","username":"username-1207_6958170162716978875","approvedUser":null,{"uid":"uid-1208_740118705315508455","username":"username-1208_2694986229407752067","approvedUser":null,{"uid":"uid-1209_3525438132159027250","username":"username-1209_4481250567040168148","approvedUser":null,{"uid":"uid-1210_7302579879694441677","username":"username-1210_6237390827868604260","approvedUser":null,{"uid":"uid-1211_-2143663206315148860","username":"username-1211_8737862624009248305","approvedUser":null,{"uid":"uid-1212_6723411512995785335","username":"username-1212_125518417770852922","approvedUser":null,{"uid":"uid-1213_157539679119339493","username":"username-1213_

945532129776754114","approvedUser":null},{\"uid\":\"uid-1214_-2127888620148840915\",\"username\":\"username-1214_6410347897302243553\",\"approvedUser\":null},{\"uid\":\"uid-1215_5763577280025505362\",\"username\":\"username-1215_-2237711036852185221\",\"approvedUser\":null},{\"uid\":\"uid-1216_4038578553369112638\",\"username\":\"username-1216_-7421962025354417257\",\"approvedUser\":null},{\"uid\":\"uid-1217_-464221293725828855\",\"username\":\"username-1217_-4965833204531848087\",\"approvedUser\":null},{\"uid\":\"uid-1218_3435004835730094458\",\"username\":\"username-1218_-6301512597356153106\",\"approvedUser\":null},{\"uid\":\"uid-1219_-5503451040469784371\",\"username\":\"username-1219_7547944351220004946\",\"approvedUser\":null},{\"uid\":\"uid-1220_7782120906154266550\",\"username\":\"username-1220_-915734404176086383\",\"approvedUser\":null},{\"uid\":\"uid-1221_-938023138081636442\",\"username\":\"username-1221_-2919268672643938465\",\"approvedUser\":null},{\"uid\":\"uid-1222_6467303696487561804\",\"username\":\"username-1222_6690795156666627404\",\"approvedUser\":null},{\"uid\":\"uid-1223_5897438603994336606\",\"username\":\"username-1223_7148058005923689251\",\"approvedUser\":null},{\"uid\":\"uid-1224_6902192985375516312\",\"username\":\"username-1224_2823680098359420307\",\"approvedUser\":null},{\"uid\":\"uid-1225_-335208494578388892\",\"username\":\"username-1225_-6223777427037406447\",\"approvedUser\":null},{\"uid\":\"uid-1226_-9218979011500727172\",\"username\":\"username-1226_526758475490761212\",\"approvedUser\":null},{\"uid\":\"uid-1227_-5357008003673285523\",\"username\":\"username-1227_4909773324936776961\",\"approvedUser\":null},{\"uid\":\"uid-1228_4941771142036173533\",\"username\":\"username-1228_5337736331291791808\",\"approvedUser\":null},{\"uid\":\"uid-1229_3211080789241309870\",\"username\":\"username-1229_8795716700784061871\",\"approvedUser\":null},{\"uid\":\"uid-1230_6679632664188230691\",\"username\":\"username-1230_-5341849463927591618\",\"approvedUser\":null},{\"uid\":\"uid-1231_-3225211286494379357\",\"username\":\"username-1231_4409258437136052540\",\"approvedUser\":null},{\"uid\":\"uid-1232_-4586800333479025987\",\"username\":\"username-1232_-3934326214998479239\",\"approvedUser\":null},{\"uid\":\"uid-1233_7762148160964316268\",\"username\":\"username-1233_-6261952715943754837\",\"approvedUser\":null},{\"uid\":\"uid-1234_-8642432071189152357\",\"username\":\"username-1234_1372207172542826901\",\"approvedUser\":null},{\"uid\":\"uid-1235_-4082489357142939414\",\"username\":\"username-1235_-3640048492994674331\",\"approvedUser\":null},{\"uid\":\"uid-1236_-1590693303257226062\",\"username\":\"username-1236_-6967343138163489443\",\"approvedUser\":null},{\"uid\":\"uid-1237_-216125469949458874\",\"username\":\"username-1237_-4417324707909754954\",\"approvedUser\":null},{\"uid\":\"uid-1238_-813061692281557636\",\"username\":\"username-1238_-1697412849244410944\",\"approvedUser\":null},{\"uid\":\"uid-1239_-9204271119793322901\",\"username\":\"username-1239_-1798458799809485491\",\"approvedUser\":null},{\"uid\":\"uid-1240_-5479936200895858827\",\"username\":\"username-1240_4919702337342753360\",\"approvedUser\":null},{\"uid\":\"uid-1241_-7807912466440427129\",\"username\":\"username-1241_-6150815375260683584\",\"approvedUser\":null},{\"uid\":\"uid-1242_6960163433142446239\",\"username\":\"username-1242_-8594735969038089534\",\"approvedUser\":null},{\"uid\":\"uid-1243_-2491831203095206258\",\"username\":\"username-1243_9143997317997976326\",\"approvedUser\":null},{\"uid\":\"uid-1244_2563818364167715327\",\"username\":\"username-1244_-9031498663176737936\",\"approvedUser\":null},{\"uid\":\"uid-1245_-7218878196333188169\",\"username\":\"username-1245_6185952247919856635\",\"approvedUser\":null},{\"uid\":\"uid-1246_-5151404509148097507\",\"username\":\"username-1246_-1872745063464943654\",\"approvedUser\":null},{\"uid\":\"uid-1247_1331259930170510589\",\"username\":\"username-1247_-7139674662750402849\",\"approvedUser\":null},{\"uid\":\"uid-1248_-1148508250036077688\",\"username\":\"username-1248_-7980350593145659104\",\"approvedUser\":null},{\"uid\":\"uid-1249_5794959510861051551\",\"username\":\"username-1249_-4184928697039853353\",\"approvedUser\":null},{\"uid\":\"uid-1250_-6121993685887988400\",\"username\":\"username-1250_6468706454975506058\",\"approvedUser\":null},{\"uid\":\"uid-1251_6562672092827185568\",\"username\":\"username-1251_4633404985071436651\",\"approvedUser\":null},{\"uid\":\"uid-1252_7503120829059009418\",\"username\":\"username-1252_1851930003093028353\",\"approvedUser\":null},{\"uid\":\"uid-1253_2303062195291781192\",\"username\":\"username-1253_4316007588261479686\",\"approvedUser\":null},{\"uid\":\"uid-1254_-1752327967127012912\",\"username\":\"username-1254_7144031944089705619\",\"approvedUser\":null},{\"uid\":\"uid-1255_5485831662381816481\",\"username\":\"username-1255_-5921851696019246726\",\"approvedUser\":null},{\"uid\":\"uid-1256_-5028468404998796866\",\"username\":\"username-1256_1152210039950704370\",\"approvedUser\":null},{\"uid\":\"uid-1257_5585752167361278714\",\"username\":\"username-1257_-2941792400376092039\",\"approvedUser\":null},{\"uid\":\"uid-1258_-4080386647301974154\",\"username\":\"username-1258_-2254704120449877514\",\"approvedUser\":null},{\"uid\":\"uid-1259_5742747346388855400\",\"username\":\"username-1259_-236480377113032675\",\"approvedUser\":null},{\"uid\":\"uid-1260_-250385146836732387\",\"username\":\"username-1260_9208125796313537302\",\"approvedUser\":null},{\"uid\":\"uid-1261_-376169935036969664\",\"username\":\"username-1261_1562463152104948065\",\"approvedUser\":null},{\"uid\":\"uid-1262_6581354379602037852\",\"username\":\"username-1262_-8450064709041743629\",\"approvedUser\":null},{\"uid\":\"uid-1263_1759743661361754015\",\"username\":\"username-1263_-622136095665249115\",\"approvedUser\":null},{\"uid\":\"uid-1264_-7934587890484952864\",\"username\":\"username-1264_-5419320307840068658\",\"approvedUser\":null},{\"uid\":\"uid-1265_588099607379452386\",\"username\":\"username-1265_-1568865338653368274\",\"approvedUser\":null},{\"uid\":\"uid-1266_2250504625131091673\",\"username\":\"username-1266_-

5560381800070896115","approvedUser":null,{"uid":"uid-1267_1843315300871025966","username":"username-1267_-6588494714313991517","approvedUser":null,{"uid":"uid-1268_-4001425935875515777","username":"username-1268_7134388186401359034","approvedUser":null,{"uid":"uid-1269_8646858137882972137","username":"username-1269_6776457526500318097","approvedUser":null,{"uid":"uid-1270_-3770475395437249143","username":"username-1270_-8204298935283388413","approvedUser":null,{"uid":"uid-1271_5814072750698762814","username":"username-1271_1051803983836038120","approvedUser":null,{"uid":"uid-1272_-3507483978468146252","username":"username-1272_-5476152318360669253","approvedUser":null,{"uid":"uid-1273_4742955395744434846","username":"username-1273_-968897523322743171","approvedUser":null,{"uid":"uid-1274_434911033183477570","username":"username-1274_-8151566542735755304","approvedUser":null,{"uid":"uid-1275_-1320868010056828358","username":"username-1275_2085683165926674566","approvedUser":null,{"uid":"uid-1276_6898507888742849912","username":"username-1276_6779124245212602611","approvedUser":null,{"uid":"uid-1277_-4416987853035755126","username":"username-1277_4059865847479931425","approvedUser":null,{"uid":"uid-1278_-1541418641704926434","username":"username-1278_-7898696523324555061","approvedUser":null,{"uid":"uid-1279_-6438500701713625911","username":"username-1279_-7656468106077563187","approvedUser":null,{"uid":"uid-1280_-5229908198856705789","username":"username-1280_3057788936216696514","approvedUser":null,{"uid":"uid-1281_4012612043494226038","username":"username-1281_5786544665679379950","approvedUser":null,{"uid":"uid-1282_-5902796761106200457","username":"username-1282_-168197977394630562","approvedUser":null,{"uid":"uid-1283_-4304008776368675372","username":"username-1283_-7211316914621177011","approvedUser":null,{"uid":"uid-1284_-4572795049145948209","username":"username-1284_-8123879360893191734","approvedUser":null,{"uid":"uid-1285_-1203313561297696918","username":"username-1285_5478244539005847446","approvedUser":null,{"uid":"uid-1286_-3217284528460671259","username":"username-1286_8810387117774294610","approvedUser":null,{"uid":"uid-1287_3720589038391750021","username":"username-1287_4542303741261727961","approvedUser":null,{"uid":"uid-1288_-8895435939394761834","username":"username-1288_4811831355391580394","approvedUser":null,{"uid":"uid-1289_2243670735708388017","username":"username-1289_-6748244293222323446","approvedUser":null,{"uid":"uid-1290_-5573586116184691134","username":"username-1290_-7942530008740907864","approvedUser":null,{"uid":"uid-1291_3083116081071368260","username":"username-1291_-1021058529432129407","approvedUser":null,{"uid":"uid-1292_5107525614742447694","username":"username-1292_-1421834622378721978","approvedUser":null,{"uid":"uid-1293_-8465601482489980205","username":"username-1293_3253432476532982395","approvedUser":null,{"uid":"uid-1294_5694149793509226149","username":"username-1294_978614678229712801","approvedUser":null,{"uid":"uid-1295_3228533008335701691","username":"username-1295_-253831172968912489","approvedUser":null,{"uid":"uid-1296_-1986649683438556621","username":"username-1296_6341838480616127036","approvedUser":null,{"uid":"uid-1297_7438380849583514777","username":"username-1297_7300583548765460057","approvedUser":null,{"uid":"uid-1298_-5252774743655903127","username":"username-1298_-54511841848791618","approvedUser":null,{"uid":"uid-1299_8823135192866076114","username":"username-1299_4578804083235486122","approvedUser":null,{"uid":"uid-1300_6117713857466681788","username":"username-1300_-3228001986693694097","approvedUser":null,{"uid":"uid-1301_3268088519714855180","username":"username-1301_-3320122937172185770","approvedUser":null,{"uid":"uid-1302_7944321951714288678","username":"username-1302_-199072183375437889","approvedUser":null,{"uid":"uid-1303_4351063344781389491","username":"username-1303_-2548037266481910644","approvedUser":null,{"uid":"uid-1304_7707300076779964324","username":"username-1304_-7723028925799719232","approvedUser":null,{"uid":"uid-1305_4551209333208993141","username":"username-1305_-2332604194372936606","approvedUser":null,{"uid":"uid-1306_-7278973410051280202","username":"username-1306_-6040575425119033177","approvedUser":null,{"uid":"uid-1307_-863263818058791226","username":"username-1307_-7365768842855488433","approvedUser":null,{"uid":"uid-1308_3475178693921638113","username":"username-1308_6373073919669071951","approvedUser":null,{"uid":"uid-1309_-7632338291682477219","username":"username-1309_-7692407006823770449","approvedUser":null,{"uid":"uid-1310_-592945020981484240","username":"username-1310_4932491263032778165","approvedUser":null,{"uid":"uid-1311_-8940554172072011","username":"username-1311_1473707293656557891","approvedUser":null,{"uid":"uid-1312_8797360653267114671","username":"username-1312_-8069259629945762179","approvedUser":null,{"uid":"uid-1313_-2179011635217133427","username":"username-1313_-2698753951457821266","approvedUser":null,{"uid":"uid-1314_-4386563194815226641","username":"username-1314_1831531120751240812","approvedUser":null,{"uid":"uid-1315_4651880223203301225","username":"username-1315_1828780265304799737","approvedUser":null,{"uid":"uid-1316_-7757996596169976186","username":"username-1316_-3345086984711639061","approvedUser":null,{"uid":"uid-1317_-4165019955862782194","username":"username-1317_5061869685253370127","approvedUser":null,{"uid":"uid-1318_3173135291620869003","username":"username-1318_7414829711627455627","approvedUser":null,{"uid":"uid-1319_3120377415921162250","username":"username-1319_-

5332793391077467326","approvedUser":null,{"uid":"uid-1320_2208609334948994590","username":"username-1320_7345307977566877051","approvedUser":null,{"uid":"uid-1321_2734804759468625193","username":"username-1321_6326174385109705020","approvedUser":null,{"uid":"uid-1322_-595079433702155554","username":"username-1322_-1566039339173164012","approvedUser":null,{"uid":"uid-1323_-2946491087763265535","username":"username-1323_-2838945837002971089","approvedUser":null,{"uid":"uid-1324_8783227648084325745","username":"username-1324_2111518492328393294","approvedUser":null,{"uid":"uid-1325_-6526391438420236952","username":"username-1325_-4263876766216350978","approvedUser":null,{"uid":"uid-1326_6759437835831184640","username":"username-1326_856576780299412296","approvedUser":null,{"uid":"uid-1327_-3355909806422500872","username":"username-1327_-7762533043670998511","approvedUser":null,{"uid":"uid-1328_7078280104256622776","username":"username-1328_-3540154802461924412","approvedUser":null,{"uid":"uid-1329_-3152996807040047470","username":"username-1329_-673337058059450323","approvedUser":null,{"uid":"uid-1330_6308101972675296330","username":"username-1330_-7055751042855951704","approvedUser":null,{"uid":"uid-1331_-4074747001512051497","username":"username-1331_6730843431805862010","approvedUser":null,{"uid":"uid-1332_-7846630700271164309","username":"username-1332_2907635904629027198","approvedUser":null,{"uid":"uid-1333_-6220221374566534223","username":"username-1333_-6111062241631652093","approvedUser":null,{"uid":"uid-1334_-2323332889675295915","username":"username-1334_8538952386221243636","approvedUser":null,{"uid":"uid-1335_5834695095764003784","username":"username-1335_-3168383923900186310","approvedUser":null,{"uid":"uid-1336_-7925819564379703996","username":"username-1336_-5586699830095278755","approvedUser":null,{"uid":"uid-1337_4697371935551718414","username":"username-1337_-1160450931262151055","approvedUser":null,{"uid":"uid-1338_-2318572410038872838","username":"username-1338_7420002409205667631","approvedUser":null,{"uid":"uid-1339_7624889383844159028","username":"username-1339_-1098242199682090911","approvedUser":null,{"uid":"uid-1340_-209863859608803268","username":"username-1340_3415338156420750211","approvedUser":null,{"uid":"uid-1341_289802647083922226","username":"username-1341_-8906445382987613273","approvedUser":null,{"uid":"uid-1342_-4101378390919435181","username":"username-1342_6542131815579130009","approvedUser":null,{"uid":"uid-1343_7411755677238900675","username":"username-1343_-6232097673727582142","approvedUser":null,{"uid":"uid-1344_-388712615751888244","username":"username-1344_-5176680767307052463","approvedUser":null,{"uid":"uid-1345_-5989510238297091866","username":"username-1345_2804490354256185879","approvedUser":null,{"uid":"uid-1346_689759238364069048","username":"username-1346_-7100348249946752334","approvedUser":null,{"uid":"uid-1347_8331103266478676063","username":"username-1347_7125808136284907885","approvedUser":null,{"uid":"uid-1348_-6471295695460087379","username":"username-1348_4465659400942641766","approvedUser":null,{"uid":"uid-1349_-3265479124753567437","username":"username-1349_1144806960583959409","approvedUser":null,{"uid":"uid-1350_222633415588295107","username":"username-1350_-8076448360682350276","approvedUser":null,{"uid":"uid-1351_-1915000760133211088","username":"username-1351_-4375854304012957005","approvedUser":null,{"uid":"uid-1352_-7425450989566839977","username":"username-1352_6093064204348593830","approvedUser":null,{"uid":"uid-1353_-1772024209806836638","username":"username-1353_-3332368579722090584","approvedUser":null,{"uid":"uid-1354_-319148300153712821","username":"username-1354_5847292985686311480","approvedUser":null,{"uid":"uid-1355_-1068328418670658657","username":"username-1355_-4855522815828815578","approvedUser":null,{"uid":"uid-1356_-1630069285559462524","username":"username-1356_7665858219248833583","approvedUser":null,{"uid":"uid-1357_-2882359591017316611","username":"username-1357_1298966522216207594","approvedUser":null,{"uid":"uid-1358_5318507555526148459","username":"username-1358_2257339664440083944","approvedUser":null,{"uid":"uid-1359_-8023431034687521962","username":"username-1359_-2164939710003073003","approvedUser":null,{"uid":"uid-1360_4620329497747482221","username":"username-1360_-3178607529429839260","approvedUser":null,{"uid":"uid-1361_-5018331712767460917","username":"username-1361_-3423294663690004565","approvedUser":null,{"uid":"uid-1362_-3351007527646548869","username":"username-1362_8528945049881347787","approvedUser":null,{"uid":"uid-1363_-6597512933723153346","username":"username-1363_6745229100110598419","approvedUser":null,{"uid":"uid-1364_-2410171112330912629","username":"username-1364_5665449952911647461","approvedUser":null,{"uid":"uid-1365_3920641787143156182","username":"username-1365_3577547707596839578","approvedUser":null,{"uid":"uid-1366_6455612735416106261","username":"username-1366_3524520526894205346","approvedUser":null,{"uid":"uid-1367_-6960709035091010689","username":"username-1367_2095772698329900016","approvedUser":null,{"uid":"uid-1368_-5462684441175048554","username":"username-1368_-2205811130729168492","approvedUser":null,{"uid":"uid-1369_-7547037759444025241","username":"username-1369_96453981981628072","approvedUser":null,{"uid":"uid-1370_8648246708393949200","username":"username-1370_3519871070210393456","approvedUser":null,{"uid":"uid-1371_5241027368772670083","username":"username-1371_554275481054342845","approvedUser":null,{"uid":"uid-1372_-2311206062596033656","username":"username-

1372_4487868432877347985","approvedUser":null},{ "uid":"uid-1373_-5661910308670594296","username":"username-1373_7104023872955147016","approvedUser":null},{ "uid":"uid-1374_1131495570682317447","username":"username-1374_6065506490081342325","approvedUser":null},{ "uid":"uid-1375_7034560438111160050","username":"username-1375_6114444015349997003","approvedUser":null},{ "uid":"uid-1376_4186066496376214170","username":"username-1376_845340032970067006","approvedUser":null},{ "uid":"uid-1377_800284515517057920","username":"username-1377_7036087534124451113","approvedUser":null},{ "uid":"uid-1378_-7068064610840700462","username":"username-1378_-4942498929929838799","approvedUser":null},{ "uid":"uid-1379_4284152959945734837","username":"username-1379_6278268390286939051","approvedUser":null},{ "uid":"uid-1380_-4297212404223823035","username":"username-1380_-5082591302380399814","approvedUser":null},{ "uid":"uid-1381_-893048024737517607","username":"username-1381_6717524553264185532","approvedUser":null},{ "uid":"uid-1382_-3174609705129405557","username":"username-1382_-6456868374832433561","approvedUser":null},{ "uid":"uid-1383_-8387245446540329993","username":"username-1383_5830855600151181614","approvedUser":null},{ "uid":"uid-1384_2051281521032674625","username":"username-1384_-4147883535805350295","approvedUser":null},{ "uid":"uid-1385_8558542932706728541","username":"username-1385_8931690439057237940","approvedUser":null},{ "uid":"uid-1386_4189623827986791818","username":"username-1386_7824257176267479318","approvedUser":null},{ "uid":"uid-1387_707558523521168670","username":"username-1387_2875737977307549029","approvedUser":null},{ "uid":"uid-1388_5673807996330670217","username":"username-1388_-7154712818149798679","approvedUser":null},{ "uid":"uid-1389_6452137945326549395","username":"username-1389_-6700090615177827776","approvedUser":null},{ "uid":"uid-1390_-2793492154092911128","username":"username-1390_6180994915064850816","approvedUser":null},{ "uid":"uid-1391_-6472782980782850859","username":"username-1391_877586565050725794","approvedUser":null},{ "uid":"uid-1392_-6520777422996562604","username":"username-1392_7514381857639532573","approvedUser":null},{ "uid":"uid-1393_-8319334529996951483","username":"username-1393_-6699565754525096239","approvedUser":null},{ "uid":"uid-1394_2809182329489855230","username":"username-1394_6362956639057861670","approvedUser":null},{ "uid":"uid-1395_7175832852389504517","username":"username-1395_7496992677591977333","approvedUser":null},{ "uid":"uid-1396_5929063591910237211","username":"username-1396_-8222505192762043482","approvedUser":null},{ "uid":"uid-1397_-4480466837817612931","username":"username-1397_-8066921069890700775","approvedUser":null},{ "uid":"uid-1398_-5532779549302107291","username":"username-1398_118858609975117769","approvedUser":null},{ "uid":"uid-1399_2589840560719564758","username":"username-1399_-5939589003870341457","approvedUser":null},{ "uid":"uid-1400_995295097232299417","username":"username-1400_-6432950963190039196","approvedUser":null},{ "uid":"uid-1401_1338089758293529730","username":"username-1401_8783457346924269321","approvedUser":null},{ "uid":"uid-1402_5916821607403064823","username":"username-1402_1049325839347512682","approvedUser":null},{ "uid":"uid-1403_-953036313158748750","username":"username-1403_-2965771546727732895","approvedUser":null},{ "uid":"uid-1404_859663953994947269","username":"username-1404_2748511651754118520","approvedUser":null},{ "uid":"uid-1405_-2160561708523741986","username":"username-1405_-464058274603209006","approvedUser":null},{ "uid":"uid-1406_-4548742192103525940","username":"username-1406_8617308079100801098","approvedUser":null},{ "uid":"uid-1407_-7239778955549982911","username":"username-1407_-5918713158256671780","approvedUser":null},{ "uid":"uid-1408_-2287864900795331367","username":"username-1408_1030780229486663235","approvedUser":null},{ "uid":"uid-1409_7662181553035579995","username":"username-1409_5252878562235486436","approvedUser":null},{ "uid":"uid-1410_122664761697478205","username":"username-1410_-643140177204701718","approvedUser":null},{ "uid":"uid-1411_-7464876007093727123","username":"username-1411_3546612444803046065","approvedUser":null},{ "uid":"uid-1412_3508014832018179406","username":"username-1412_6396814422241723495","approvedUser":null},{ "uid":"uid-1413_-6870400699439198976","username":"username-1413_-2774893523165271312","approvedUser":null},{ "uid":"uid-1414_-1433433434936747292","username":"username-1414_5592490601900221962","approvedUser":null},{ "uid":"uid-1415_-6428941281059481249","username":"username-1415_-6166179510565746510","approvedUser":null},{ "uid":"uid-1416_5965759346657684957","username":"username-1416_-571506839552741691","approvedUser":null},{ "uid":"uid-1417_6712044240879991064","username":"username-1417_-6938105282396520215","approvedUser":null},{ "uid":"uid-1418_-5869890714790078567","username":"username-1418_-3895159699878784339","approvedUser":null},{ "uid":"uid-1419_-8199305312133486978","username":"username-1419_5402087524716297966","approvedUser":null},{ "uid":"uid-1420_3527521466502189182","username":"username-1420_-7553754421374065027","approvedUser":null},{ "uid":"uid-1421_-850661666662488278","username":"username-1421_-8717561026215856898","approvedUser":null},{ "uid":"uid-1422_4493181363928239221","username":"username-1422_-2786860635033759726","approvedUser":null},{ "uid":"uid-1423_-1801006220775886730","username":"username-1423_7281555244828644536","approvedUser":null},{ "uid":"uid-1424_1468269240789018475","username":"username-1424_8933599643314734832","approvedUser":null},{ "uid":"uid-1425_-5936278235610145791","username":"username-1425_-

4294243902615070816","approvedUser":null},{ "uid":"uid-1426_-928577767200972335","username":"username-1426_-
6533860097179589445","approvedUser":null},{ "uid":"uid-1427_-1341663393780290514","username":"username-1427_-
7705848729047181408","approvedUser":null},{ "uid":"uid-1428_3524948820566158621","username":"username-
1428_3582895621306107327","approvedUser":null},{ "uid":"uid-1429_-4106724030234042204","username":"username-1429_-
280864425468528766","approvedUser":null},{ "uid":"uid-1430_-3877496741949667257","username":"username-
1430_295117217149614441","approvedUser":null},{ "uid":"uid-1431_-5894590634876871985","username":"username-
1431_8121759161147333495","approvedUser":null},{ "uid":"uid-1432_1135361889733127245","username":"username-1432_-
2282229603135388980","approvedUser":null},{ "uid":"uid-1433_-1288392228982549923","username":"username-
1433_8031176669252694419","approvedUser":null},{ "uid":"uid-1434_1133172847518681263","username":"username-
1434_4617952662480202014","approvedUser":null},{ "uid":"uid-1435_1892763299239188931","username":"username-1435_-
8003947481447036338","approvedUser":null},{ "uid":"uid-1436_3881601917834678259","username":"username-
1436_8483429976780298296","approvedUser":null},{ "uid":"uid-1437_6942666024665412278","username":"username-1437_-
6340314812576481020","approvedUser":null},{ "uid":"uid-1438_7631821958363042850","username":"username-
1438_8950572547527414040","approvedUser":null},{ "uid":"uid-1439_-3656634423461498190","username":"username-
1439_819013125668123766","approvedUser":null},{ "uid":"uid-1440_-6676116932159802551","username":"username-
1440_443456014081354338","approvedUser":null},{ "uid":"uid-1441_2037992275360954231","username":"username-
1441_6111039214294470472","approvedUser":null},{ "uid":"uid-1442_-8964893151894793222","username":"username-1442_-
5081409571505495845","approvedUser":null},{ "uid":"uid-1443_-7700925020894844026","username":"username-1443_-
5711198834848735103","approvedUser":null},{ "uid":"uid-1444_616930804575613384","username":"username-
1444_599857948145989613","approvedUser":null},{ "uid":"uid-1445_-3290422394065937752","username":"username-
1445_5683276674539668493","approvedUser":null},{ "uid":"uid-1446_-3230451272398636594","username":"username-1446_-
439143244575773273","approvedUser":null},{ "uid":"uid-1447_520316120360113768","username":"username-1447_-
6575701763326993602","approvedUser":null},{ "uid":"uid-1448_-3596044928035653847","username":"username-
1448_5955934172853348794","approvedUser":null},{ "uid":"uid-1449_-9118209193842414956","username":"username-1449_-
438976664061684409","approvedUser":null},{ "uid":"uid-1450_4245602272961318777","username":"username-1450_-
8805353852307476228","approvedUser":null},{ "uid":"uid-1451_-5775718564538768961","username":"username-
1451_2647292286935779264","approvedUser":null},{ "uid":"uid-1452_5517932024745758564","username":"username-1452_-
117277471393307927","approvedUser":null},{ "uid":"uid-1453_-1374333941561550398","username":"username-
1453_8517654026573977381","approvedUser":null},{ "uid":"uid-1454_9189877947779876627","username":"username-1454_-
7475960917956412515","approvedUser":null},{ "uid":"uid-1455_-6232811890392504265","username":"username-1455_-
7916580974845743273","approvedUser":null},{ "uid":"uid-1456_-3365365354961619276","username":"username-1456_-
7666863286831148837","approvedUser":null},{ "uid":"uid-1457_6355875521698778877","username":"username-
1457_6394319905100188697","approvedUser":null},{ "uid":"uid-1458_-3310361894765835530","username":"username-1458_-
572985360587562359","approvedUser":null},{ "uid":"uid-1459_-6293822470963986249","username":"username-
1459_1546987573530923412","approvedUser":null},{ "uid":"uid-1460_5523998329938389392","username":"username-
1460_3559571247225928498","approvedUser":null},{ "uid":"uid-1461_3883558911832094028","username":"username-1461_-
2416094905211276077","approvedUser":null},{ "uid":"uid-1462_-1575793227354600006","username":"username-
1462_4744046936815919235","approvedUser":null},{ "uid":"uid-1463_-3956290951881994648","username":"username-
1463_8011390409262707784","approvedUser":null},{ "uid":"uid-1464_-4072598255576894287","username":"username-
1464_7836164087363572686","approvedUser":null},{ "uid":"uid-1465_-5277804482430383110","username":"username-1465_-
3468707932154683323","approvedUser":null},{ "uid":"uid-1466_7894505534263597624","username":"username-1466_-
1686440105993641334","approvedUser":null},{ "uid":"uid-1467_-6206127814153449292","username":"username-1467_-
3811521216503090300","approvedUser":null},{ "uid":"uid-1468_-282202825093135086","username":"username-
1468_5872544450878752464","approvedUser":null},{ "uid":"uid-1469_7335854449270912655","username":"username-1469_-
8532349316145742944","approvedUser":null},{ "uid":"uid-1470_-6757784390114991223","username":"username-
1470_3562354794578499552","approvedUser":null},{ "uid":"uid-1471_-6570599734575287484","username":"username-1471_-
6077559857852984935","approvedUser":null},{ "uid":"uid-1472_-5746589405623126038","username":"username-1472_-
6610790026345264486","approvedUser":null},{ "uid":"uid-1473_4503636421687381203","username":"username-1473_-
2345104335324388780","approvedUser":null},{ "uid":"uid-1474_-6392664665261419257","username":"username-1474_-
514943178585406970","approvedUser":null},{ "uid":"uid-1475_3359158038576520704","username":"username-
1475_4736096760661051161","approvedUser":null},{ "uid":"uid-1476_-5981658508365092173","username":"username-
1476_3764732800229486540","approvedUser":null},{ "uid":"uid-1477_-1585824008616465711","username":"username-1477_-
3566505739284009453","approvedUser":null},{ "uid":"uid-1478_-5563316973903519159","username":"username-1478_-

8761028402764282692","approvedUser":null},{\"uid\":\"uid-1479_-6015462137220168430\",\"username\":\"username-1479_-4159840265379310892\",\"approvedUser\":null},{\"uid\":\"uid-1480_8151595258396707622\",\"username\":\"username-1480_1448310749107581256\",\"approvedUser\":null},{\"uid\":\"uid-1481_1186228526173225169\",\"username\":\"username-1481_3700349514585212622\",\"approvedUser\":null},{\"uid\":\"uid-1482_1190600124827309867\",\"username\":\"username-1482_-869661201039428091\",\"approvedUser\":null},{\"uid\":\"uid-1483_5515582038989453538\",\"username\":\"username-1483_-6152806109325702693\",\"approvedUser\":null},{\"uid\":\"uid-1484_-306887024380434628\",\"username\":\"username-1484_-2261324041391602642\",\"approvedUser\":null},{\"uid\":\"uid-1485_-1474625636823932069\",\"username\":\"username-1485_-453663550574857547\",\"approvedUser\":null},{\"uid\":\"uid-1486_-2362967394410567741\",\"username\":\"username-1486_-1365927111301566943\",\"approvedUser\":null},{\"uid\":\"uid-1487_-5956475761088006631\",\"username\":\"username-1487_-5444401996809738369\",\"approvedUser\":null},{\"uid\":\"uid-1488_5433793404790242605\",\"username\":\"username-1488_8825552506147170271\",\"approvedUser\":null},{\"uid\":\"uid-1489_-8735393958075999430\",\"username\":\"username-1489_8797076980502849082\",\"approvedUser\":null},{\"uid\":\"uid-1490_-3957311335989051284\",\"username\":\"username-1490_-3725280270655383664\",\"approvedUser\":null},{\"uid\":\"uid-1491_7102309255653471649\",\"username\":\"username-1491_6523145085213744465\",\"approvedUser\":null},{\"uid\":\"uid-1492_8019121741590268787\",\"username\":\"username-1492_-2889625508685269507\",\"approvedUser\":null},{\"uid\":\"uid-1493_6574297950247499126\",\"username\":\"username-1493_-377073554575240178\",\"approvedUser\":null},{\"uid\":\"uid-1494_3801031229522859453\",\"username\":\"username-1494_6532930086733922069\",\"approvedUser\":null},{\"uid\":\"uid-1495_-6708749189383979869\",\"username\":\"username-1495_-641008802619350238\",\"approvedUser\":null},{\"uid\":\"uid-1496_5117933206870352071\",\"username\":\"username-1496_-3225105617528999320\",\"approvedUser\":null},{\"uid\":\"uid-1497_3685487792406795096\",\"username\":\"username-1497_-8581149959281656543\",\"approvedUser\":null},{\"uid\":\"uid-1498_-6097033134707886958\",\"username\":\"username-1498_-5298657776543538257\",\"approvedUser\":null},{\"uid\":\"uid-1499_-6011288630662304633\",\"username\":\"username-1499_-6556650695979219197\",\"approvedUser\":null},{\"uid\":\"uid-1500_5656506565845232674\",\"username\":\"username-1500_1988698909878934338\",\"approvedUser\":null},{\"uid\":\"uid-1501_-9036289100715165080\",\"username\":\"username-1501_-6996698079038717787\",\"approvedUser\":null},{\"uid\":\"uid-1502_8161266757851664897\",\"username\":\"username-1502_2799404493107087008\",\"approvedUser\":null},{\"uid\":\"uid-1503_-5941152409265122569\",\"username\":\"username-1503_-2559867618552119994\",\"approvedUser\":null},{\"uid\":\"uid-1504_1324517973343554748\",\"username\":\"username-1504_-6031669044057409301\",\"approvedUser\":null},{\"uid\":\"uid-1505_-6747216436142578575\",\"username\":\"username-1505_5539460469202136074\",\"approvedUser\":null},{\"uid\":\"uid-1506_-8244136541741786267\",\"username\":\"username-1506_-2261176027221394583\",\"approvedUser\":null},{\"uid\":\"uid-1507_4337236539793979867\",\"username\":\"username-1507_264164617776857210\",\"approvedUser\":null},{\"uid\":\"uid-1508_3064559731342402063\",\"username\":\"username-1508_8097056034820533764\",\"approvedUser\":null},{\"uid\":\"uid-1509_-7365036242671349893\",\"username\":\"username-1509_-284753471523446014\",\"approvedUser\":null},{\"uid\":\"uid-1510_-4703478859105368490\",\"username\":\"username-1510_113937386869356491\",\"approvedUser\":null},{\"uid\":\"uid-1511_-5937808221499820195\",\"username\":\"username-1511_7988424878727041394\",\"approvedUser\":null},{\"uid\":\"uid-1512_-2939058312102519852\",\"username\":\"username-1512_-647954633669408913\",\"approvedUser\":null},{\"uid\":\"uid-1513_7259091456182209999\",\"username\":\"username-1513_1513_635664697523877758\",\"approvedUser\":null},{\"uid\":\"uid-1514_1705257538534613165\",\"username\":\"username-1514_5138574363267638215\",\"approvedUser\":null},{\"uid\":\"uid-1515_-2993460562241307163\",\"username\":\"username-1515_-1121243065893771458\",\"approvedUser\":null},{\"uid\":\"uid-1516_-7050172017209308668\",\"username\":\"username-1516_6767196521240169548\",\"approvedUser\":null},{\"uid\":\"uid-1517_-2115348854619859466\",\"username\":\"username-1517_-7811536400754120273\",\"approvedUser\":null},{\"uid\":\"uid-1518_4891643392453451907\",\"username\":\"username-1518_5072664118843337117\",\"approvedUser\":null},{\"uid\":\"uid-1519_253291333209140252\",\"username\":\"username-1519_-8038425098351878353\",\"approvedUser\":null},{\"uid\":\"uid-1520_-8903689114856706856\",\"username\":\"username-1520_-2979889934759637507\",\"approvedUser\":null},{\"uid\":\"uid-1521_-2622340198199651656\",\"username\":\"username-1521_6950072821892728837\",\"approvedUser\":null},{\"uid\":\"uid-1522_-5871644785355082702\",\"username\":\"username-1522_2818409274531431888\",\"approvedUser\":null},{\"uid\":\"uid-1523_3933589419099970797\",\"username\":\"username-1523_4463652264112315607\",\"approvedUser\":null},{\"uid\":\"uid-1524_-4456385676500597560\",\"username\":\"username-1524_7775279754446895138\",\"approvedUser\":null},{\"uid\":\"uid-1525_-3991001873037921504\",\"username\":\"username-1525_-7525582438700353743\",\"approvedUser\":null},{\"uid\":\"uid-1526_708133134503498322\",\"username\":\"username-1526_3211256514678773536\",\"approvedUser\":null},{\"uid\":\"uid-1527_-779060625813735554\",\"username\":\"username-1527_-4318823179516524027\",\"approvedUser\":null},{\"uid\":\"uid-1528_937121063609557646\",\"username\":\"username-1528_-1457869181857013604\",\"approvedUser\":null},{\"uid\":\"uid-1529_5461204548921670262\",\"username\":\"username-1529_-5738920933626893786\",\"approvedUser\":null},{\"uid\":\"uid-1530_-859899637610264132\",\"username\":\"username-1530_-8410675736391553395\",\"approvedUser\":null},{\"uid\":\"uid-1531_3869062470562561851\",\"username\":\"username-1531_-

1101846979107199285","approvedUser":null},{"uid":"uid-1532_-2981932087752214749","username":"username-1532_2009847657226213261","approvedUser":null},{"uid":"uid-1533_-8808325626113404861","username":"username-1533_-756526394828115437","approvedUser":null},{"uid":"uid-1534_-5289335315303411318","username":"username-1534_6070353708476669785","approvedUser":null},{"uid":"uid-1535_-4044475888289358794","username":"username-1535_2522984980842125691","approvedUser":null},{"uid":"uid-1536_-1999951287671956415","username":"username-1536_-7159002355254134955","approvedUser":null},{"uid":"uid-1537_-5398808825889714096","username":"username-1537_3679015979724705385","approvedUser":null},{"uid":"uid-1538_-6656610073029974292","username":"username-1538_8748565912568057605","approvedUser":null},{"uid":"uid-1539_-6304735058654515479","username":"username-1539_-7603494765447807323","approvedUser":null},{"uid":"uid-1540_-6821581261816624675","username":"username-1540_2322288075387122837","approvedUser":null},{"uid":"uid-1541_-3694847749468464474","username":"username-1541_-8946015265255621414","approvedUser":null},{"uid":"uid-1542_-7148934094890838029","username":"username-1542_3505256868497212754","approvedUser":null},{"uid":"uid-1543_-2356587918017576776","username":"username-1543_-12665892339025638","approvedUser":null},{"uid":"uid-1544_-4881266249083217100","username":"username-1544_-2688444689406071249","approvedUser":null},{"uid":"uid-1545_-1246260123190287474","username":"username-1545_8289582373971134807","approvedUser":null},{"uid":"uid-1546_-6322400045084728832","username":"username-1546_4188420832207420481","approvedUser":null},{"uid":"uid-1547_-4328114374167610057","username":"username-1547_5802140561754440171","approvedUser":null},{"uid":"uid-1548_-80776724938418622","username":"username-1548_5762463048484994371","approvedUser":null},{"uid":"uid-1549_-597763295391317103","username":"username-1549_-1808208353341989359","approvedUser":null},{"uid":"uid-1550_-6905947285722094763","username":"username-1550_-3516043581444526061","approvedUser":null},{"uid":"uid-1551_-694038774637388735","username":"username-1551_-5328080796717877143","approvedUser":null},{"uid":"uid-1552_-8695785524840429491","username":"username-1552_-1957363351056857805","approvedUser":null},{"uid":"uid-1553_-8587028436367685768","username":"username-1553_-8516737250305633416","approvedUser":null},{"uid":"uid-1554_-7942668589370910646","username":"username-1554_4519208030397017929","approvedUser":null},{"uid":"uid-1555_-8541861856498583705","username":"username-1555_3018482214503035430","approvedUser":null},{"uid":"uid-1556_-3905379022403553166","username":"username-1556_-2896661527250353249","approvedUser":null},{"uid":"uid-1557_-4354859479348052044","username":"username-1557_-6458188681105031106","approvedUser":null},{"uid":"uid-1558_-8014603154105988472","username":"username-1558_8916399390504544932","approvedUser":null},{"uid":"uid-1559_-1377541454973677317","username":"username-1559_2817051108162669297","approvedUser":null},{"uid":"uid-1560_-7826486470367374342","username":"username-1560_-8719984561840414615","approvedUser":null},{"uid":"uid-1561_-935999154458160298","username":"username-1561_-6060306630257637037","approvedUser":null},{"uid":"uid-1562_-4213116941345731656","username":"username-1562_2580780424850851108","approvedUser":null},{"uid":"uid-1563_-5250685412042169039","username":"username-1563_6538097338910127325","approvedUser":null},{"uid":"uid-1564_-6882610144902191125","username":"username-1564_2877987169451081136","approvedUser":null},{"uid":"uid-1565_-7289557014819719711","username":"username-1565_-3466068821745208855","approvedUser":null},{"uid":"uid-1566_-7936865430964956124","username":"username-1566_-4495351361609849844","approvedUser":null},{"uid":"uid-1567_-4795173001363437503","username":"username-1567_7200015590725453881","approvedUser":null},{"uid":"uid-1568_-3677863693491654925","username":"username-1568_6226442516593297368","approvedUser":null},{"uid":"uid-1569_-7470346102458606192","username":"username-1569_-9083605625158849679","approvedUser":null},{"uid":"uid-1570_-6441575022864760812","username":"username-1570_5841021157333810907","approvedUser":null},{"uid":"uid-1571_-4286975070435199564","username":"username-1571_-534801695703559272","approvedUser":null},{"uid":"uid-1572_-5972273699377982437","username":"username-1572_2238152625554769279","approvedUser":null},{"uid":"uid-1573_-8170019632061309358","username":"username-1573_-6441788821782977125","approvedUser":null},{"uid":"uid-1574_-3683494853402146527","username":"username-1574_5325443220616490835","approvedUser":null},{"uid":"uid-1575_-246496313649393365","username":"username-1575_282372529214696394","approvedUser":null},{"uid":"uid-1576_-7259713768127429444","username":"username-1576_7721672546988238728","approvedUser":null},{"uid":"uid-1577_-2060522135680341006","username":"username-1577_-3553134497755619751","approvedUser":null},{"uid":"uid-1578_-6861513802791305509","username":"username-1578_-5349710176251051659","approvedUser":null},{"uid":"uid-1579_-3446305423060313714","username":"username-1579_-7810253708738757663","approvedUser":null},{"uid":"uid-1580_-4071647484092417944","username":"username-1580_-994349435107960556","approvedUser":null},{"uid":"uid-1581_-7710041471413255725","username":"username-1581_-2332584718881410596","approvedUser":null},{"uid":"uid-1582_-6535889668542292535","username":"username-1582_1625977672674493314","approvedUser":null},{"uid":"uid-1583_-1040705661042730874","username":"username-1583_-6293641299998010068","approvedUser":null},{"uid":"uid-1584_-8718090219848757312","username":"username-

1584_1620502868521897858","approvedUser":null},{\"uid\":\"uid-1585_4063234698530737016\",\"username\":\"username-1585_-2156952145122727276\",\"approvedUser\":null},{\"uid\":\"uid-1586_2547213060356234930\",\"username\":\"username-1586_2489375366832756219\",\"approvedUser\":null},{\"uid\":\"uid-1587_-7885890095334503770\",\"username\":\"username-1587_4682112348999647037\",\"approvedUser\":null},{\"uid\":\"uid-1588_-1446292326848330300\",\"username\":\"username-1588_5885563417043638902\",\"approvedUser\":null},{\"uid\":\"uid-1589_-8507069672792051976\",\"username\":\"username-1589_-1464172893098716877\",\"approvedUser\":null},{\"uid\":\"uid-1590_8710479644810817839\",\"username\":\"username-1590_-7307685270860514399\",\"approvedUser\":null},{\"uid\":\"uid-1591_1313121138586426136\",\"username\":\"username-1591_3363766125445564390\",\"approvedUser\":null},{\"uid\":\"uid-1592_-747567691862560978\",\"username\":\"username-1592_1604058976211071950\",\"approvedUser\":null},{\"uid\":\"uid-1593_934084600764919798\",\"username\":\"username-1593_6426684848440605866\",\"approvedUser\":null},{\"uid\":\"uid-1594_6398179709020679301\",\"username\":\"username-1594_-4607065591886698124\",\"approvedUser\":null},{\"uid\":\"uid-1595_-4462575658102337211\",\"username\":\"username-1595_-7921588120039746761\",\"approvedUser\":null},{\"uid\":\"uid-1596_4456834577561321720\",\"username\":\"username-1596_5404831905412681645\",\"approvedUser\":null},{\"uid\":\"uid-1597_2702161809887345147\",\"username\":\"username-1597_-3634837329443970519\",\"approvedUser\":null},{\"uid\":\"uid-1598_-3837525340039836721\",\"username\":\"username-1598_3398457630377121463\",\"approvedUser\":null},{\"uid\":\"uid-1599_6926796261146042581\",\"username\":\"username-1599_813924088329448321\",\"approvedUser\":null},{\"uid\":\"uid-1600_4407168219121360984\",\"username\":\"username-1600_-4225092744332612493\",\"approvedUser\":null},{\"uid\":\"uid-1601_1658082301658770641\",\"username\":\"username-1601_-6094718561272322782\",\"approvedUser\":null},{\"uid\":\"uid-1602_-3148968695325392283\",\"username\":\"username-1602_2493644805687872745\",\"approvedUser\":null},{\"uid\":\"uid-1603_6451018235184970161\",\"username\":\"username-1603_-518954217612013227\",\"approvedUser\":null},{\"uid\":\"uid-1604_-3661209871413480758\",\"username\":\"username-1604_7480080205681779394\",\"approvedUser\":null},{\"uid\":\"uid-1605_-855691402035630977\",\"username\":\"username-1605_-166256268700670139\",\"approvedUser\":null},{\"uid\":\"uid-1606_1223713317401316320\",\"username\":\"username-1606_7339918482332219341\",\"approvedUser\":null},{\"uid\":\"uid-1607_2496996925632944131\",\"username\":\"username-1607_6294853334882126724\",\"approvedUser\":null},{\"uid\":\"uid-1608_413872715047710213\",\"username\":\"username-1608_-3953656778856319622\",\"approvedUser\":null},{\"uid\":\"uid-1609_4790416094730137935\",\"username\":\"username-1609_317194735145064581\",\"approvedUser\":null},{\"uid\":\"uid-1610_-6077264638747263974\",\"username\":\"username-1610_8848356452304762210\",\"approvedUser\":null},{\"uid\":\"uid-1611_3596921040715125812\",\"username\":\"username-1611_1795498052698060575\",\"approvedUser\":null},{\"uid\":\"uid-1612_-3334391657955465213\",\"username\":\"username-1612_-8522882072073756227\",\"approvedUser\":null},{\"uid\":\"uid-1613_9197019688664520650\",\"username\":\"username-1613_-2687696293415205104\",\"approvedUser\":null},{\"uid\":\"uid-1614_-5313380903053471957\",\"username\":\"username-1614_-5204148615326358038\",\"approvedUser\":null},{\"uid\":\"uid-1615_7643213348149883628\",\"username\":\"username-1615_509571376173520760\",\"approvedUser\":null},{\"uid\":\"uid-1616_-2817867165816117507\",\"username\":\"username-1616_-4918213359617642773\",\"approvedUser\":null},{\"uid\":\"uid-1617_-86068959593788231\",\"username\":\"username-1617_-1505885642995339430\",\"approvedUser\":null},{\"uid\":\"uid-1618_2076255657077913964\",\"username\":\"username-1618_7151236973666483174\",\"approvedUser\":null},{\"uid\":\"uid-1619_80282649637139733\",\"username\":\"username-1619_-8770915889145240034\",\"approvedUser\":null},{\"uid\":\"uid-1620_-7246264709166861972\",\"username\":\"username-1620_239907847290453669\",\"approvedUser\":null},{\"uid\":\"uid-1621_-9137322408283680474\",\"username\":\"username-1621_2803518622924295186\",\"approvedUser\":null},{\"uid\":\"uid-1622_-5705773457022847114\",\"username\":\"username-1622_1275681607727123479\",\"approvedUser\":null},{\"uid\":\"uid-1623_3514998608660293591\",\"username\":\"username-1623_-3547685943904017692\",\"approvedUser\":null},{\"uid\":\"uid-1624_-2566046675709338104\",\"username\":\"username-1624_602835222884158669\",\"approvedUser\":null},{\"uid\":\"uid-1625_6314346192184615512\",\"username\":\"username-1625_3879650509512418857\",\"approvedUser\":null},{\"uid\":\"uid-1626_716268146336128547\",\"username\":\"username-1626_-3803644621637165699\",\"approvedUser\":null},{\"uid\":\"uid-1627_4438569415350334619\",\"username\":\"username-1627_8080372591745262554\",\"approvedUser\":null},{\"uid\":\"uid-1628_-5198274930471623733\",\"username\":\"username-1628_7930746028947885955\",\"approvedUser\":null},{\"uid\":\"uid-1629_-4446379274874981927\",\"username\":\"username-1629_1361194371065734354\",\"approvedUser\":null},{\"uid\":\"uid-1630_4111034214204451467\",\"username\":\"username-1630_-4806861529224431269\",\"approvedUser\":null},{\"uid\":\"uid-1631_-6807441979353796340\",\"username\":\"username-1631_3796884465317176657\",\"approvedUser\":null},{\"uid\":\"uid-1632_151825146986080750\",\"username\":\"username-1632_-8646216994694764760\",\"approvedUser\":null},{\"uid\":\"uid-1633_8342918950793789810\",\"username\":\"username-1633_-3794455993522840196\",\"approvedUser\":null},{\"uid\":\"uid-1634_-8461681373645467129\",\"username\":\"username-1634_9100052092058532657\",\"approvedUser\":null},{\"uid\":\"uid-1635_7435321157861125904\",\"username\":\"username-1635_-5084555865826174502\",\"approvedUser\":null},{\"uid\":\"uid-1636_-3716104209369425942\",\"username\":\"username-1636_-4819860077247033762\",\"approvedUser\":null},{\"uid\":\"uid-1637_196260489277413165\",\"username\":\"username-

1637_6126667179759281112","approvedUser":null},{ "uid":"uid-1638_-3087758707924676048","username":"username-1638_-2825945310366064831","approvedUser":null},{ "uid":"uid-1639_-189865184621874386","username":"username-1639_-5750965600860043847","approvedUser":null},{ "uid":"uid-1640_-3234160354561746296","username":"username-1640_-3604767893550897669","approvedUser":null},{ "uid":"uid-1641_-9060864997403980463","username":"username-1641_-5709736360126637607","approvedUser":null},{ "uid":"uid-1642_-1274329611398562016","username":"username-1642_-3960973403762222585","approvedUser":null},{ "uid":"uid-1643_-3801841330238719163","username":"username-1643_-136806626574740057","approvedUser":null},{ "uid":"uid-1644_-6496729655783560560","username":"username-1644_-962083645801781567","approvedUser":null},{ "uid":"uid-1645_-7173275309711483496","username":"username-1645_-3601883774318608466","approvedUser":null},{ "uid":"uid-1646_-4154972116543822895","username":"username-1646_-169192815796345163","approvedUser":null},{ "uid":"uid-1647_-4163067257664537229","username":"username-1647_-5506743016498994867","approvedUser":null},{ "uid":"uid-1648_-3865278506477097108","username":"username-1648_-6092306948157201044","approvedUser":null},{ "uid":"uid-1649_-968053966381123900","username":"username-1649_-1964542291337264184","approvedUser":null},{ "uid":"uid-1650_-6282403597339567498","username":"username-1650_-4997371538337733878","approvedUser":null},{ "uid":"uid-1651_-2809960096323852772","username":"username-1651_-383283008974555336","approvedUser":null},{ "uid":"uid-1652_-259802420738016124","username":"username-1652_-5602594261676363218","approvedUser":null},{ "uid":"uid-1653_-2947859891802457900","username":"username-1653_-4210577665200968409","approvedUser":null},{ "uid":"uid-1654_-9160211323479177807","username":"username-1654_-5180881311483829766","approvedUser":null},{ "uid":"uid-1655_-4572282846065019190","username":"username-1655_-629830981445434859","approvedUser":null},{ "uid":"uid-1656_-7049649623184452243","username":"username-1656_-1588525982620525212","approvedUser":null},{ "uid":"uid-1657_-5967720268055510138","username":"username-1657_-8688927448308176942","approvedUser":null},{ "uid":"uid-1658_-4601450568939810158","username":"username-1658_-8959830664114920640","approvedUser":null},{ "uid":"uid-1659_-5754206978774739","username":"username-1659_-2366164789517519046","approvedUser":null},{ "uid":"uid-1660_-7879793711624958608","username":"username-1660_-16324864184664270","approvedUser":null},{ "uid":"uid-1661_-1577780230117213112","username":"username-1661_-8161894563299679204","approvedUser":null},{ "uid":"uid-1662_-3216959190697796184","username":"username-1662_-7738621393832063496","approvedUser":null},{ "uid":"uid-1663_-8987363726736308835","username":"username-1663_-3234127675338909782","approvedUser":null},{ "uid":"uid-1664_-7485465793110213195","username":"username-1664_-7527178286490076682","approvedUser":null},{ "uid":"uid-1665_-7674213742281290325","username":"username-1665_-3494234014466165121","approvedUser":null},{ "uid":"uid-1666_-1412797177192063226","username":"username-1666_-8182206082874526641","approvedUser":null},{ "uid":"uid-1667_-6885427952626374704","username":"username-1667_-486574382728917207","approvedUser":null},{ "uid":"uid-1668_-2975296477905974651","username":"username-1668_-5859691514331843155","approvedUser":null},{ "uid":"uid-1669_-7441927420279623588","username":"username-1669_-5803887793625186988","approvedUser":null},{ "uid":"uid-1670_-7528949781742137838","username":"username-1670_-7125046836239608875","approvedUser":null},{ "uid":"uid-1671_-4112322140219367614","username":"username-1671_-1989177855116191214","approvedUser":null},{ "uid":"uid-1672_-2910010584404579227","username":"username-1672_-6343619297401546966","approvedUser":null},{ "uid":"uid-1673_-5502022512040592876","username":"username-1673_-5498546940591522902","approvedUser":null},{ "uid":"uid-1674_-2666215094319858040","username":"username-1674_-537309370490777958","approvedUser":null},{ "uid":"uid-1675_-3999496374061810397","username":"username-1675_-5136015660005308792","approvedUser":null},{ "uid":"uid-1676_-2922784227847359987","username":"username-1676_-3630674625292484198","approvedUser":null},{ "uid":"uid-1677_-1693935636757377031","username":"username-1677_-8100250092241485236","approvedUser":null},{ "uid":"uid-1678_-1715495133732862789","username":"username-1678_-6774679626482737244","approvedUser":null},{ "uid":"uid-1679_-5020041062094375452","username":"username-1679_-294582923958070102","approvedUser":null},{ "uid":"uid-1680_-5909958502528457483","username":"username-1680_-1502002076520876794","approvedUser":null},{ "uid":"uid-1681_-429212155866932991","username":"username-1681_-810801777422169632","approvedUser":null},{ "uid":"uid-1682_-3813500097857145902","username":"username-1682_-5490452576347533977","approvedUser":null},{ "uid":"uid-1683_-4563435344327461652","username":"username-1683_-426966136396740293","approvedUser":null},{ "uid":"uid-1684_-3317932560415267086","username":"username-1684_-7467615355078670095","approvedUser":null},{ "uid":"uid-1685_-962034528580024323","username":"username-1685_-6023637894250858378","approvedUser":null},{ "uid":"uid-1686_-7928001912592176199","username":"username-1686_-3039393490015269742","approvedUser":null},{ "uid":"uid-1687_-5221903929381212471","username":"username-1687_-1401668420915804923","approvedUser":null},{ "uid":"uid-1688_-4284972343540860623","username":"username-1688_-2062167069639997333","approvedUser":null},{ "uid":"uid-1689_-2894547558857855070","username":"username-1689_-8976121955245874268","approvedUser":null},{ "uid":"uid-1690_-6171305372633955282","username":"username-1690_-

6702032913473094469","approvedUser":null},{"uid":"uid-1691_-2415962126348449300","username":"username-1691_-536862767301353979","approvedUser":null},{"uid":"uid-1692_-4631393826323833018","username":"username-1692_-7168433889121279815","approvedUser":null},{"uid":"uid-1693_-4504669107271564905","username":"username-1693_-8141466803938809506","approvedUser":null},{"uid":"uid-1694_-3065073868800711370","username":"username-1694_-8220849411628132721","approvedUser":null},{"uid":"uid-1695_-5402266966723916068","username":"username-1695_-2808055084342270835","approvedUser":null},{"uid":"uid-1696_-3884126855334670931","username":"username-1696_-1163443966916532515","approvedUser":null},{"uid":"uid-1697_-829756940336763257","username":"username-1697_-6941192003609554389","approvedUser":null},{"uid":"uid-1698_-4130876808884969154","username":"username-1698_-3389581581688062058","approvedUser":null},{"uid":"uid-1699_-186803275451545073","username":"username-1699_-4282473844259640128","approvedUser":null},{"uid":"uid-1700_-9065714646485908861","username":"username-1700_-7961108602487478433","approvedUser":null},{"uid":"uid-1701_-6143456242234051083","username":"username-1701_-7983535438833423801","approvedUser":null},{"uid":"uid-1702_-5785757194047271300","username":"username-1702_-4038293849385100273","approvedUser":null},{"uid":"uid-1703_-1904430822409511251","username":"username-1703_-3049236712121299737","approvedUser":null},{"uid":"uid-1704_-8001563417968454255","username":"username-1704_-7366266353166455065","approvedUser":null},{"uid":"uid-1705_-5490724894493968792","username":"username-1705_-6019139807753124348","approvedUser":null},{"uid":"uid-1706_-6889069888380358688","username":"username-1706_-7801796263854046590","approvedUser":null},{"uid":"uid-1707_-2216425555400437908","username":"username-1707_-505117582716421265","approvedUser":null},{"uid":"uid-1708_-8048974588113252476","username":"username-1708_-2070576066533514447","approvedUser":null},{"uid":"uid-1709_-6276513179669090984","username":"username-1709_-7039912064714105635","approvedUser":null},{"uid":"uid-1710_-7516995902267646357","username":"username-1710_-5086651750678870775","approvedUser":null},{"uid":"uid-1711_-1630833978398627099","username":"username-1711_-4521061596689443951","approvedUser":null},{"uid":"uid-1712_-7570411938761137116","username":"username-1712_-6276723644138184634","approvedUser":null},{"uid":"uid-1713_-8566273164073229315","username":"username-1713_-5682884824677802912","approvedUser":null},{"uid":"uid-1714_-4005884276239190145","username":"username-1714_-5978108905298279677","approvedUser":null},{"uid":"uid-1715_-4868844174082448038","username":"username-1715_-506130972588340873","approvedUser":null},{"uid":"uid-1716_-2249358550249289993","username":"username-1716_-1757318771293261641","approvedUser":null},{"uid":"uid-1717_-6523117128666030260","username":"username-1717_-1345834499343060003","approvedUser":null},{"uid":"uid-1718_-5017266249212689843","username":"username-1718_-3188016163679302065","approvedUser":null},{"uid":"uid-1719_-5864147858513269208","username":"username-1719_-8992814763205182444","approvedUser":null},{"uid":"uid-1720_-333554287732693828","username":"username-1720_-8058037461871471967","approvedUser":null},{"uid":"uid-1721_-7843551445671841790","username":"username-1721_-8600254535158217630","approvedUser":null},{"uid":"uid-1722_-16543854661579806","username":"username-1722_-433247889940835311","approvedUser":null},{"uid":"uid-1723_-2742290679843725285","username":"username-1723_-3886349532742965114","approvedUser":null},{"uid":"uid-1724_-4175468144754056173","username":"username-1724_-5305445188593961446","approvedUser":null},{"uid":"uid-1725_-1518145464031378808","username":"username-1725_-1109806052294946626","approvedUser":null},{"uid":"uid-1726_-6086612175209536219","username":"username-1726_-2721868055491619554","approvedUser":null},{"uid":"uid-1727_-4833346419187482742","username":"username-1727_-6942312674450402808","approvedUser":null},{"uid":"uid-1728_-1839071981809388502","username":"username-1728_-4003230997031251933","approvedUser":null},{"uid":"uid-1729_-8286389101648933086","username":"username-1729_-5010855250898801279","approvedUser":null},{"uid":"uid-1730_-8078553124371825875","username":"username-1730_-5288647209000756387","approvedUser":null},{"uid":"uid-1731_-4810405498694161549","username":"username-1731_-1178428334712338399","approvedUser":null},{"uid":"uid-1732_-1757668210072509968","username":"username-1732_-2187174824341142344","approvedUser":null},{"uid":"uid-1733_-9202249036724472260","username":"username-1733_-8921070656066834183","approvedUser":null},{"uid":"uid-1734_-2754351198438855033","username":"username-1734_-8663981558264073619","approvedUser":null},{"uid":"uid-1735_-1682668627790849285","username":"username-1735_-3047827531807853372","approvedUser":null},{"uid":"uid-1736_-593422587276303348","username":"username-1736_-8432292039461838019","approvedUser":null},{"uid":"uid-1737_-5137747213384601053","username":"username-1737_-7989781996892101835","approvedUser":null},{"uid":"uid-1738_-3622377532089369063","username":"username-1738_-6566174925846839218","approvedUser":null},{"uid":"uid-1739_-6877468242688200086","username":"username-1739_-8767842933077518092","approvedUser":null},{"uid":"uid-1740_-6152975195806914523","username":"username-1740_-67426465552795428","approvedUser":null},{"uid":"uid-1741_-9168708853066048439","username":"username-1741_-5953398428294009339","approvedUser":null},{"uid":"uid-1742_-8959830182254131107","username":"username-1742_-5497171214473162051","approvedUser":null},{"uid":"uid-1743_-6602544827567305167","username":"username-1743_-

6899127028937572351","approvedUser":null,{"uid":"uid-1744_8305157210968122333","username":"username-1744_5351070850637195142","approvedUser":null,{"uid":"uid-1745_-483147866380780984","username":"username-1745_6685054750623757065","approvedUser":null,{"uid":"uid-1746_-8165043668186966148","username":"username-1746_-4786522923800792117","approvedUser":null,{"uid":"uid-1747_-3207217180929074536","username":"username-1747_1650599895955008380","approvedUser":null,{"uid":"uid-1748_-698280198281998732","username":"username-1748_5556662099533348061","approvedUser":null,{"uid":"uid-1749_-7926686618612054811","username":"username-1749_-5554147940373101914","approvedUser":null,{"uid":"uid-1750_894127346745434687","username":"username-1750_3894379853086814377","approvedUser":null,{"uid":"uid-1751_4833274314019446043","username":"username-1751_8195539530421576502","approvedUser":null,{"uid":"uid-1752_8662168622355363417","username":"username-1752_5797974510020657345","approvedUser":null,{"uid":"uid-1753_-3959570611266474648","username":"username-1753_-1109534335426227774","approvedUser":null,{"uid":"uid-1754_-5265729609249934140","username":"username-1754_6105095191519163100","approvedUser":null,{"uid":"uid-1755_-8902932338217456960","username":"username-1755_-693796239980051135","approvedUser":null,{"uid":"uid-1756_-536025614175479163","username":"username-1756_4403897759653904724","approvedUser":null,{"uid":"uid-1757_-613823705140950694","username":"username-1757_-4996558199892291697","approvedUser":null,{"uid":"uid-1758_-2958180941732113371","username":"username-1758_-6142107165468669018","approvedUser":null,{"uid":"uid-1759_4383478119799824689","username":"username-1759_-3284101092323994132","approvedUser":null,{"uid":"uid-1760_2770746840279722544","username":"username-1760_7490479866460264182","approvedUser":null,{"uid":"uid-1761_-5696357455833406844","username":"username-1761_1496744976133233021","approvedUser":null,{"uid":"uid-1762_-5091019144402616903","username":"username-1762_9028813190914186888","approvedUser":null,{"uid":"uid-1763_1897898972696074838","username":"username-1763_-7349751039517192246","approvedUser":null,{"uid":"uid-1764_8061010445847844053","username":"username-1764_-3286728497519332280","approvedUser":null,{"uid":"uid-1765_-964017156461246738","username":"username-1765_7941041018624543550","approvedUser":null,{"uid":"uid-1766_6364879352516048667","username":"username-1766_-157609554590568125","approvedUser":null,{"uid":"uid-1767_3404334491352792152","username":"username-1767_-5769475944520223829","approvedUser":null,{"uid":"uid-1768_-871122510571709909","username":"username-1768_-4913321955378088404","approvedUser":null,{"uid":"uid-1769_1020108947126933731","username":"username-1769_-7295814684564123965","approvedUser":null,{"uid":"uid-1770_-1300937017100689884","username":"username-1770_9070132016625695704","approvedUser":null,{"uid":"uid-1771_-585034824321119838","username":"username-1771_7831601028954577056","approvedUser":null,{"uid":"uid-1772_-5415977656807450069","username":"username-1772_-6271222471029410878","approvedUser":null,{"uid":"uid-1773_9059149708071755499","username":"username-1773_3234166569804098621","approvedUser":null,{"uid":"uid-1774_7849345132988250347","username":"username-1774_7161825792822868104","approvedUser":null,{"uid":"uid-1775_-2876566547332894502","username":"username-1775_1801253655526603770","approvedUser":null,{"uid":"uid-1776_-6416465368008104688","username":"username-1776_5063838340781575788","approvedUser":null,{"uid":"uid-1777_1017029277421622223","username":"username-1777_-1865718631713973732","approvedUser":null,{"uid":"uid-1778_2080886850633958859","username":"username-1778_-2140738545071017005","approvedUser":null,{"uid":"uid-1779_2104413821032209644","username":"username-1779_1239620699389137226","approvedUser":null,{"uid":"uid-1780_8741076337465068369","username":"username-1780_-3882377232710106679","approvedUser":null,{"uid":"uid-1781_862056110455912088","username":"username-1781_3640228271423524113","approvedUser":null,{"uid":"uid-1782_-2974188509906826404","username":"username-1782_-4564029632906912644","approvedUser":null,{"uid":"uid-1783_-3331865626507737858","username":"username-1783_-6465960155773742495","approvedUser":null,{"uid":"uid-1784_3835572240149212584","username":"username-1784_-4892173439866047555","approvedUser":null,{"uid":"uid-1785_6989267370483145359","username":"username-1785_-1977636543910623217","approvedUser":null,{"uid":"uid-1786_-7019621270175078281","username":"username-1786_-6172891603417361534","approvedUser":null,{"uid":"uid-1787_-1183053248969704324","username":"username-1787_-7961091475166589103","approvedUser":null,{"uid":"uid-1788_1275206996801850638","username":"username-1788_5060903874106557542","approvedUser":null,{"uid":"uid-1789_-3574950644386695364","username":"username-1789_-6487541604364325819","approvedUser":null,{"uid":"uid-1790_-7502037719363607498","username":"username-1790_-2814940026469696472","approvedUser":null,{"uid":"uid-1791_-4785177619328105716","username":"username-1791_1159363952728882794","approvedUser":null,{"uid":"uid-1792_1694889753712401083","username":"username-1792_8434255827458367016","approvedUser":null,{"uid":"uid-1793_-5678432407669229462","username":"username-1793_-3244009412460639964","approvedUser":null,{"uid":"uid-1794_-8007119969127744016","username":"username-1794_1540586910426205165","approvedUser":null,{"uid":"uid-1795_7360940839403345050","username":"username-1795_-355857629156307908","approvedUser":null,{"uid":"uid-1796_8313075465404680234","username":"username-

1796_846309371904492961","approvedUser":null},{ "uid":"uid-1797_-153958401285747167","username":"username-1797_-
5428713434448270242","approvedUser":null},{ "uid":"uid-1798_-6744235968236176063","username":"username-
1798_6328472219193596563","approvedUser":null},{ "uid":"uid-1799_1367899870515493196","username":"username-
1799_8164431220426042266","approvedUser":null},{ "uid":"uid-1800_-7883083284134258926","username":"username-1800_-
3371300858668652108","approvedUser":null},{ "uid":"uid-1801_6903442790852885674","username":"username-
1801_5728057761494381028","approvedUser":null},{ "uid":"uid-1802_-6840876597222374913","username":"username-1802_-
8425657312000052709","approvedUser":null},{ "uid":"uid-1803_1992805117864334734","username":"username-1803_-
1727435265840274028","approvedUser":null},{ "uid":"uid-1804_7491286929401637229","username":"username-
1804_8866683347744420737","approvedUser":null},{ "uid":"uid-1805_-6192001062135190386","username":"username-
1805_4261882787286504423","approvedUser":null},{ "uid":"uid-1806_6677159569017279556","username":"username-
1806_4065224745837108934","approvedUser":null},{ "uid":"uid-1807_1417932842117249543","username":"username-1807_-
5258090169477207684","approvedUser":null},{ "uid":"uid-1808_8436115486170976627","username":"username-1808_-
7241133381469841814","approvedUser":null},{ "uid":"uid-1809_438247243707185814","username":"username-
1809_4068222025989962455","approvedUser":null},{ "uid":"uid-1810_-151186142332221462","username":"username-
1810_2375984155067596564","approvedUser":null},{ "uid":"uid-1811_-4240892496955378784","username":"username-
1811_8477956324946435297","approvedUser":null},{ "uid":"uid-1812_8486061488372851103","username":"username-
1812_3993405703360110607","approvedUser":null},{ "uid":"uid-1813_-8253049387233146933","username":"username-
1813_8835425518813367654","approvedUser":null},{ "uid":"uid-1814_8867680835007834890","username":"username-
1814_679201711298008008","approvedUser":null},{ "uid":"uid-1815_7276276448372448131","username":"username-1815_-
4795366295878259818","approvedUser":null},{ "uid":"uid-1816_-3873613965908329304","username":"username-1816_-
1653923189514172335","approvedUser":null},{ "uid":"uid-1817_4318872797988047222","username":"username-1817_-
2843516440102993788","approvedUser":null},{ "uid":"uid-1818_7876507600580712699","username":"username-
1818_4413310853508009953","approvedUser":null},{ "uid":"uid-1819_-8234061995037366504","username":"username-
1819_2848074611948722090","approvedUser":null},{ "uid":"uid-1820_-211995123601922680","username":"username-1820_-
5410645647571058350","approvedUser":null},{ "uid":"uid-1821_7369937878935705888","username":"username-
1821_6590490511202903907","approvedUser":null},{ "uid":"uid-1822_-5864694433959606440","username":"username-1822_-
8187983832389228382","approvedUser":null},{ "uid":"uid-1823_-3604632861082503159","username":"username-1823_-
7377799619504371344","approvedUser":null},{ "uid":"uid-1824_140114353389883255","username":"username-1824_-
752236281961658510","approvedUser":null},{ "uid":"uid-1825_2550885876967621778","username":"username-1825_-
2325146804902968460","approvedUser":null},{ "uid":"uid-1826_-7905914636732041351","username":"username-1826_-
3004486998913407606","approvedUser":null},{ "uid":"uid-1827_-2933678875596051394","username":"username-
1827_4023810873817407352","approvedUser":null},{ "uid":"uid-1828_753024145816645104","username":"username-
1828_236916220371251539","approvedUser":null},{ "uid":"uid-1829_6086506800426327509","username":"username-1829_-
4842299802830760857","approvedUser":null},{ "uid":"uid-1830_-6303278718994631945","username":"username-
1830_4447576982401445979","approvedUser":null},{ "uid":"uid-1831_-7294273715154162207","username":"username-1831_-
8055770702216970605","approvedUser":null},{ "uid":"uid-1832_966450028882461866","username":"username-
1832_8993927165663444949","approvedUser":null},{ "uid":"uid-1833_4115079218471832370","username":"username-1833_-
6239289407939336146","approvedUser":null},{ "uid":"uid-1834_5633510267969903531","username":"username-1834_-
3496423218272082666","approvedUser":null},{ "uid":"uid-1835_-2291414959747333285","username":"username-1835_-
3118187803831623214","approvedUser":null},{ "uid":"uid-1836_6791831335043430303","username":"username-
1836_6191395359044103705","approvedUser":null},{ "uid":"uid-1837_-4848930487562660046","username":"username-
1837_5837981999419206394","approvedUser":null},{ "uid":"uid-1838_-948607617463895121","username":"username-
1838_2024059470239310332","approvedUser":null},{ "uid":"uid-1839_2645142256518324052","username":"username-
1839_4132491463300704902","approvedUser":null},{ "uid":"uid-1840_-1072004848598607096","username":"username-
1840_1226945941547237761","approvedUser":null},{ "uid":"uid-1841_-1601838912118521698","username":"username-
1841_5718945510739130171","approvedUser":null},{ "uid":"uid-1842_5925019111038801243","username":"username-1842_-
959756162431035506","approvedUser":null},{ "uid":"uid-1843_-3597219928366485323","username":"username-1843_-
3854666151131402175","approvedUser":null},{ "uid":"uid-1844_2942103870073717597","username":"username-1844_-
3983089403235608147","approvedUser":null},{ "uid":"uid-1845_3485089068235606655","username":"username-1845_-
8696377212998457949","approvedUser":null},{ "uid":"uid-1846_2160987160109839176","username":"username-
1846_8384789061760538765","approvedUser":null},{ "uid":"uid-1847_-29156398283181400","username":"username-
1847_5865514739243815123","approvedUser":null},{ "uid":"uid-1848_5451139447112258905","username":"username-1848_-
1277457216965525377","approvedUser":null},{ "uid":"uid-1849_7829138901465123871","username":"username-

1849_708581214597176611","approvedUser":null},{ "uuid":"uuid-1850_-3117507255588882352","username":"username-1850_509873094769977850","approvedUser":null},{ "uuid":"uuid-1851_-7098690156589204330","username":"username-1851_1945532895303614540","approvedUser":null},{ "uuid":"uuid-1852_-293558400845064717","username":"username-1852_-2158821527879697897","approvedUser":null},{ "uuid":"uuid-1853_-2704052063436357371","username":"username-1853_160605171028386539","approvedUser":null},{ "uuid":"uuid-1854_-4363159847687147637","username":"username-1854_-1377191694388423915","approvedUser":null},{ "uuid":"uuid-1855_5471523605051658791","username":"username-1855_-5280498757786593025","approvedUser":null},{ "uuid":"uuid-1856_5873645662278384228","username":"username-1856_-3826120093203999530","approvedUser":null},{ "uuid":"uuid-1857_7137096237742643451","username":"username-1857_6668645390741465709","approvedUser":null},{ "uuid":"uuid-1858_7942014587883445330","username":"username-1858_-8882143178829288095","approvedUser":null},{ "uuid":"uuid-1859_8550867605543875396","username":"username-1